Cross-Platform Mobile Application Using Firebase, Flutter and Dart Programming Language

Sahil Hussain Siddiqui[#1], Sharad Sharma[#2], Soumya Jaiswal[#3], Kulyash[#4]

[#]Department of Computer Science and Engineering, Noida Institute of Engineering and Technology
Knowledge Park II, Greater Noida, Uttar Pradesh, India

[1]siddiquisahil02@gmail.com
[2]sharad.sharma1999@gmail.com
[3]soumya.jais1406@gmail.com
[4]kulyashdahiya77@gmail.com

***Abstract*— In this research, we present a mobile application for Android and iOS devices. We used Dart language and Flutter framework for developing the Front-end of our society management application and used Firebase as a Cloud Back-end service for authentication, persisted data storage, push notifications and analytics. We also used BLoC architecture as our state management system. We intend to create a platform where tenets of a society can easily communicate with their management, by using our application, residents can lodge complaints, raise their concerns in society chat room, schedule their meeting easily, pay their rent and provide entry access to their visitors and delivery services with support for a digital noticeboard.**

***Keywords*— Flutter Framework, Dart Language, Firebase, BLoC architecture, Front-end, Back-end, Push notifications.**

I. INTRODUCTION

Flutter is one of the most dynamic and popular products that was developed by Google in the year 2017. In the year 2018 Flutter was introduced to mobile app developers and from there it received huge response and popularity for its ability to develop feature-rich app interfaces. It also manages to gather a huge community in a very short period of time. Flutter is a great tool from Google for creating cross-platform applications which can be deployed to the web, desktop and mobile. Flutter is basically an open-source UI development kit to develop cross platforms apps from a single code base. The earlier version of Flutter was known as codename “Sky” and first ran on the Android operating system. It used Dart language and at 2015. Dart developer summit with the stated intent of being able to render consistently at 120 frames per second. Flutter underpins utilizing shared bundles contributed by different designers to the Flutter and Dart biological systems. This permit to rapidly manufacture the application without creating everything sans preparation. Existing bundles empower many use categories, for instance, making network requests, Custom route/course dealing, utilizing native Android/iOS system services using platform channels and utilizing third-party SDKs. We must use Flutter because:

1) Flutter utilizations Dart, a quick OOP with a few valuable highlights.

2) Flutter has it is very own user interface parts, alongside a motor to deliver them on the iOS and android stages. Most of those user interface sections, directly out of the container, fit in with the rules of Material Structure.

3) Flutter doesn’t use any bridge to convert its widgets into native components which saves time and increases speed.

II. EXISTING SYSTEM

Examination of most recent cross-stage mobile application improvement approaches which are at present accessible in the market. A portion of the cross-stage versatile application advancement approaches are React Native, Ionic and so on.

The recognize approaches that utilize a run-time condition and those that create stage explicit applications from a typical code base at incorporate time. The last mentioned, generator-based classification incorporates display driven arrangements and cross-gathering. Up to now, there are no generation prepared arrangements of this class. Consequently, till these sorts of methodologies are focuses on cross-stage arrangements that consolidate the source code of an application with a runtime domain. The most noticeable mixture structure till date for cross stage application improvement is React Native, a JavaScript library developed by Facebook, which lets you build application for android and iOS using a bride which compile its JavaScript and CSS (for styling) code into native components for android and iOS, resulting in a huge performance degradation and slower render time.

III. SYSTEM ARCHITECTURE

This application developed with flutter for Android and iOS devices for all in one society management solution contains several features will help users to manage their society services and establish a clear, responsive communication to there housing management. This app consists of multiple features each dedicated to fulfilling a specific user's requirement. The key features of the app are:

1. Visitor/Vehicles entry management
2. Daily services management
3. Hire a Personnel
4. Complaint Centre
5. Digital Noticeboard
6. Pending Bill management and payment centre
7. Emergency numbers board
8. Associations meeting planner

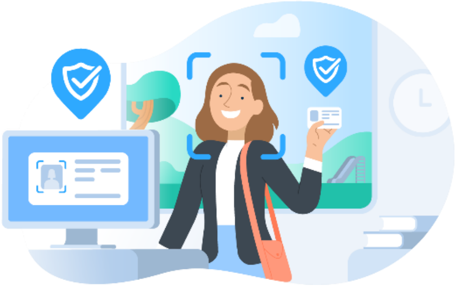
Google Firebase is a Google-backed application development software that enables developers to develop iOS, Android and Web apps. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment. Firebase offers several services, including:

- *Analytics* – Google Analytics for Firebase offers free, unlimited reporting on as many as 500 separate events. Analytics presents data about user behaviour in iOS and Android apps, enabling better decision-making about improving performance and app marketing.
- *Authentication* – Firebase Authentication makes it easy for developers to build secure authentication systems and enhances the sign-in and onboarding experience for users. This feature offers a complete identity solution, supporting email and password accounts, phone auth, as well as Google, Facebook, GitHub, Twitter login and more.
- *Cloud messaging* – Firebase Cloud Messaging (FCM) is a cross-platform messaging tool that lets companies reliably receive and deliver messages on iOS, Android and the web at no cost.
- *Realtime database* – the Firebase Realtime Database is a cloud-hosted NoSQL database that enables data to be stored and synced between users in real time. The data is synced across all clients in real time and is still available when an app goes offline.
- *Crashlytics* – Firebase Crashlytics is a real-time crash reporter that helps developers track, prioritize and fix stability issues that reduce the quality of their apps. With crashlytics, developers spend less time organizing and troubleshooting crashes and more time building features for their apps.
- *Test lab* – Firebase Test Lab is a cloud-based app-testing infrastructure. With one operation, developers can test their iOS or Android apps across a variety of devices and device configurations. They can see the results, including videos, screenshots and logs, in the Firebase console.

IV. PROPOSED SYSTEM

The home page of the app will display the details for current visitors and services logged into the society using your permit, your current outstanding bill, complaint tracking and any unread notices. It will also allow user users to navigate to:

1. Visitor management section:



This module will allow users to accept or deny the permission for entry of any visitor/vehicles in the premises. Whenever a visitor shows up at the society gates, the management will ask for the information from the guest (like where, purpose, how long etc) and then ping a notification to the registered user of that estate to allow or deny their passage into the society.

2. Daily services management:

This module will allow users to see if their daily services personnel (maid, cook, driver, laundry etc) have checked into the society or not. Once a personnel checks into the society, he/she will be asked to show their ID badge/number to be marked "checked into the premise" for that day and vice-versa for check-outs. Respective users can also see this data.

3. Hire a Personnel:

This module will allow users to hire a service offered by a registered vendor (like carpenter, appliance repair, salon) on one-time or monthly basis. Residents can also rate and leave comments on completion of the job too, which will be visible to the other residents also.

4. Complaint Centre:

This module will allow users to send their complaints to the responsible department of the management and will keep track of it and users can also upload a picture/video of the issue with the complaint.

5. Digital Noticeboard:

This module will manage the task to deliver all the issued notices or announcements to the user via push notification and noticeboard section in the application in real time.

6. Payment Centre:

This module will allow user to pay their management, utilities and outstanding bill from the app using a payment gateway integrated in the application itself. Credit and debit cards will be accepted.

7. Emergency Board:

This section will contain all the necessary nearest emergency phone numbers like Ambulance, Blood Bank, Fire Station, Police etc.

8. Associations Planner:

This section will help society member s association to log, pre-plan and keep track of there meeting. Valid Users can generate an event for monthly meeting of association and can add agendas and minutes of the last meetings.

V. ADVANTAGES OF THE PROPOSED SYSTEM

This application offers an organised, managed and responsive platform for users to interact with their residential management. Using Flutter, we can easily deploy application on Android and iOS platform (which holds 72.84% and 26.34% of Mobile operating systems market share respectively) without writing separate code, any updates can be pushed using only one codebase and flutter will compile for both platforms. Using Firebase cloud services as our backend will allow us to make our application be scalable, flexible and reactive. Firebase offers Realtime cloud services such as Cloud Firestore, Authentication and Cloud messaging (Push Notifications). Our application will keep track of every entry and transaction for every resident and digitalize the way a management works in our society making it easier, robust and faster for our users and management, and will reduce the amount of paperwork done in our current system, ultimately reducing the use of paper. Our Visitor management system guarantees that no unauthorised person/vehicle will enter the residential premises making it a better and secure residential estate.

VI. FUTURE SCOPE

This project can be further improved by:

- Making the application available in multiple regional languages.
- Allowing in-app internet calling facility.
- Public chatroom for discussing/interacting with other residents and event planning.
- Releasing a separate web application for management.

VII. CONCLUSION

The main objective of this project is to provide residents with a complete solution to manage their services from a touch on the screen. This project aims to revolutionize society management system by digitalizing everything resulting in a better, well-managed, fast and secure system.

VIII. REFERENCES

[1] Madhuram, Ashu Kumar and Pandyamanian “Cross Platform Development using Flutter”, IJESC Research article, volume 20, Issue No. 4, 2019.

[2] (2021) Flutter website. [Online]. Available: https://docs.flutter.dev/

[3] (2021) Firebase website. [Online]. Available: https://firebase.google.com/

[4] Linda Rosencrance. (2019) TechTarget webpage on Google Firebase. [Online]. Available: https://searchmobilecomputing.techtarget.com/definition/Google-Firebase