Class, A Cross Platform Flutter Application for Online Classes

Ashutosh Kumar[1] , Ashutosh Tripathi[2] , Harsh Tibrewal[3] , Ishu Raj[4]

Mr Sureshkumar M[5]

[1,2,3,4] *Students, Department of Information Science and Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India*

[5] *Associate Professor, Department of Information Science and Engineering, Dayananda Sagar College of Engineering Bangalore, Karnataka, India*

Abstract - *A mobile application for teachers, students and the administration of the department which will provide a single platform to conduct online classes and implement features like attendance tracking, reports generation and single dashboard for students to access all course related resources. The administration will be able to upload important notices and announcements which will be notified to the students. The application will be cross platform and available for both IOS and Android users. The same application will be usable for the teachers as well students although with a different dashboard experience. This application will be a complete package for college departments to manage all kinds of course resources from administration as well as students end.*

*Key Words***: Cross Platform Mobile Application Development, Flutter, Cloud Database**

1. INTRODUCTION

Due to the COVID-19 pandemic a sudden boom in the online technology sector was seen in various industries. Including apps such as Google Classroom, Zoom, Slack, etc.

Where applications such as Google Classroom provide features such as class discussions, video calls, forms posting etc. but there are some limitations such where a structured notes management system for students is missing. Also features for teachers for attendance management are out of its scope.

Our project is a cross platform application for students as well as teachers to fulfil all these requirement of institute specific features for an e portal for online classes. A structured flow where teachers can post class related resources will be available to the students. Teachers can mark attendance for every class which can then be downloaded in various file formats.

To sum up, although there are tools available for online classes, they are not configured according to institution specific requirements and class patterns, our aim is to simplify the user experience for online classes for teachers as well as the students.

The performance of mobile devices, especially smart phones, has been quickly improved for the last few years. Most users take advantage of highly efficient smart phones, and consume the contents in the smart phones longer time than other devices usage time. Also, as fast as the growth of websites was during the early 2000s similarly greater has been the growth for mobile applications now. Also, now the industry is evolving to develop cross platform applications instead of native applications.

Cross platform frameworks like Flutter and React are cheaper and fast in terms of the development process. Flutter is a purely widget-based development frameworks and creating complex and new age intuitive UI is very simple and fast process with Flutter. Online resource management and class conduction applications are available but design wise they are not as appropriate. Making easier user experience and structured sorting of all resources is what is also required. Cloud based storage of all files and features such as handwriting recognition and plagiarism checker are required as more and more assignments are given online.

1.1 Application

This kind of an application is the prime requirementfor this situation of pandemic where offline classes arenot an option. It will help the students organize alltheir resources related to classes and track theircurrent progress in terms of attendance and marks. Teachers will also have better data organizationsoptions such as marks and attendance and generatereports according to that. They can allow students forvarious subjects and verify for the same.

1.2 Registration

The registration process will be done by the department or the system admin. The credentials of each user i.e. students and teacher. Once the admin registers the user, they are assigned a unique code and their corresponding credentials for login. The basic profile only includes the student's name but rest of the details can be entered by the user once they login for the first time in the application.

1.3 Time Table

The user will be able to see their upcoming classes and this will include the links of their class conduction if any.

1.4 Department Notices

The system admin will be able to post notices and announcements for students and teachers those will be auto notified on each person's phone by push notifications. This way a centralized and more systematic way of notifying the user is created.

1.5 Marks Reporting

All the marks and exams conducted for a student will be recorded online though this system and the student has access to previous marks scored in a particular course in the application always. The dashboard will contain all the reports related to previous exams and other assignments taken by the student. Similarly, the teacher will have access to all data of the students enrolled in their course.

1.6 Course Wise Resource Posting

As notes, eBooks and other video resources posted by the teacher will be centralized and available to the student in the application itself.

1.7 File Download and Upload

The notes and other resources shared between the application users can be uploaded by these users in PDF file format. This can be then downloaded by other users also.

2. LITERATURE REVIEW

In the recent years' education technology has evolved majorly and taken a steep upside movement. Various new ERP systems with more and more functionalities are getting implemented. Android OS and the IOS OS also have seen various application launches in this segment. Although, the development process of both this platform is entirely different. Hence, two sets of code is often maintained in the organizations that is updated simultaneously when new features are added. New technologies and frameworks like React Native and Flutter have started gaining popularity as the development process through these become less expensive and less time consuming. Also, organizations have the capability of maintaining only one code base throughout. These frameworks have also opened up the opportunity of better and faster complex animated user interface. There fast grooving community has also led to creating and publishing new and extensive packages which can be used by other developers.

A recent IDC (www.idc.org) survey [1] suggests that 70% of organizations are currently deploying at least one mobile application, with more than a third of these companies deploying multiple mobile applications. In the last few years, there has been a dramatic shift in information technology. This includes various ways in which files can be shared and stored. Android OS is the latest mobile OS that has been gradually taking over the ever-expanding market share.

The performance of mobile devices, mainly smartphones, has improved rapidly over the past years. Many users use high-performance smartphones, and use content on smart phones longer than other devices. A centralized institute specific system that could be used to manage all the educational resources and assignments in one place was the idea that implemented this system. A cloud-based storage system has now become the most basic functionality of any requirement phase. Thus, this system is completely cloud based where all the data is stored in NoSQL format in Firebase Servers.

A comparison between Native applications and Flutter application was done during the research to see if the performance will not cause app crashes in a cross platform application or not. Flutter is a new tool and it continues to grow incredibly fast. Conclusions are drawn that a Flutter application can compete with a native application when it comes to CPU performance, but is not as developed in the animation area. Flutter does not require complex code for

creating a simple application and uses significantly less lines of code in development compared to native. The final conclusion is that Flutter is best to use when building smaller to medium-sized applications, but has a potential to grow to overcome its current drawbacks in the animation department.[2]

The most popular way of building Flutter application is using the BloC (Business Logic Component) design pattern. Briefly, the BloC pattern allows the exposition of data from a data layer to the UI in a reactive way using data streams. Similarly, events from the UI layer are delegated back to the data layer.[3]

2.1 Existing System:

The existing system for conducting such classes include applications such as Google Classroom, Zoom, Slack and Microsoft Teams. All these applications are generalized for online group video calls and group sharing of resources. Google Classroom utilizes Google Drive and other products like Meet to deliver a system for file sharing and video conferencing. It

chat like interface and streams for notes sharing and notice broadcasting. Thus, while these applications solve the problem of online video-based class conduction but fail to deliver a wholesome package to students and teachers for online institutional management features.

2.3 Disadvantages of Existing System

☐ All the features are not present on a single platform

☐ A complete package and a singleton point for all the actors and users is not present

☐ Highly generalized, does not implement institute specific functionalities

3. PROPOSED MODEL

The Class will be a single platform for all the users on the department of the college with role specific functionalities. For instance, a teacher will be able to
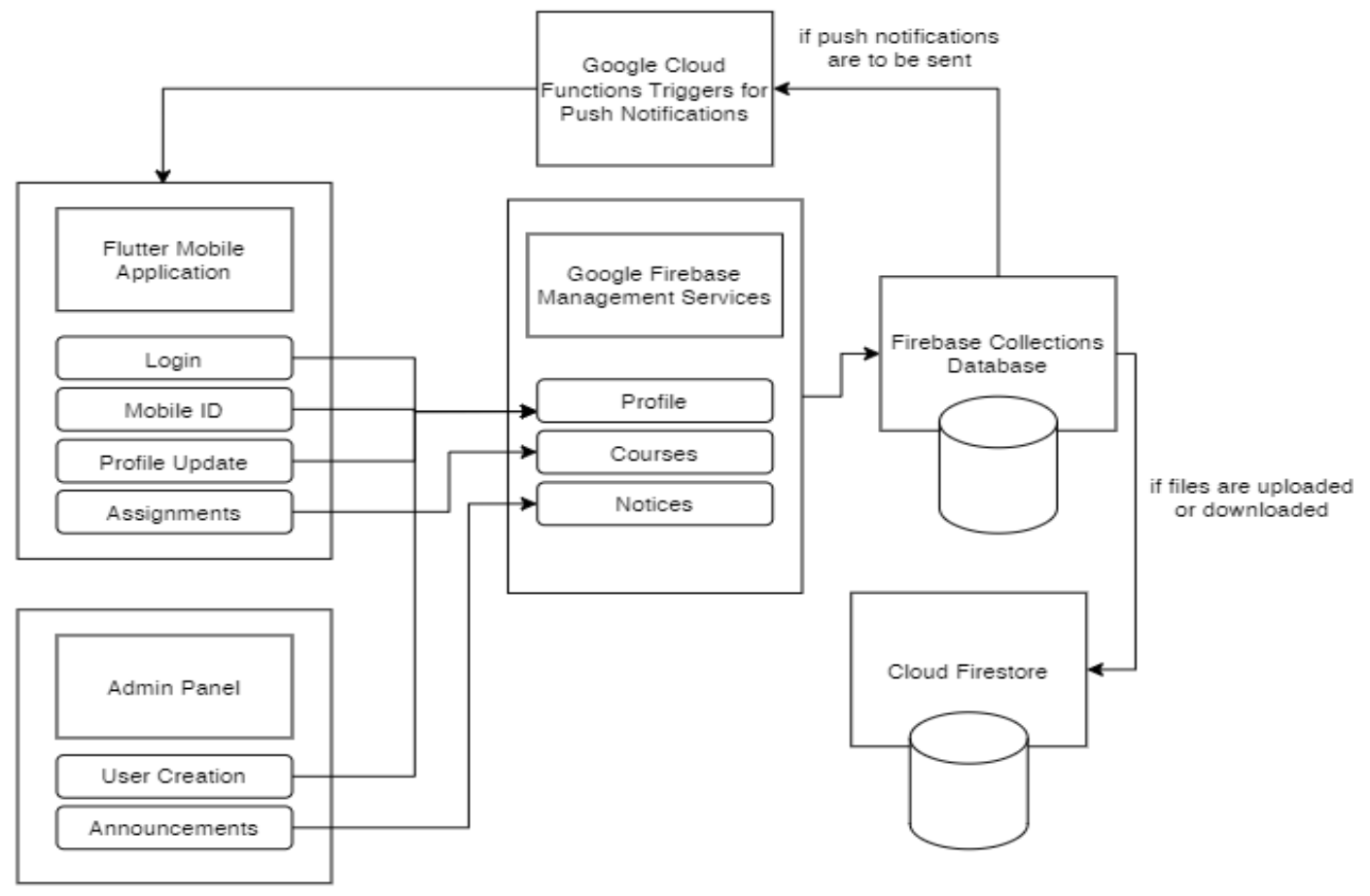

Fig: Layering and Folder Structure in Client Side App

add notes that will be shown on the dashboard of the student. Also, department administration will be able to upload notices and announcements that will be notified to teachers and students accordingly. All dashboard related functionalities will be present for teachers as well for the students. The dashboard for students will be different. All this will be available on a cross platform mobile application.

3.1 Architecture

The architecture of the application is such that all the network and cloud-based callings are done through Firebase Services. Accordingly, then database changes are made in the Firebase Collections and if files are uploaded then Firebase Storage is used to upload images, files etc. Firebase Storage provides a storage bucket which is free to a certain limit and bandwidth for Spark Plan users. Google Cloud Functions are created to send Push Notifications to a device. Triggers re written in GCP which are called whenever

easily. Whenever there are minor changes in the data layer it will not affect the presentation layer or the logic layer.
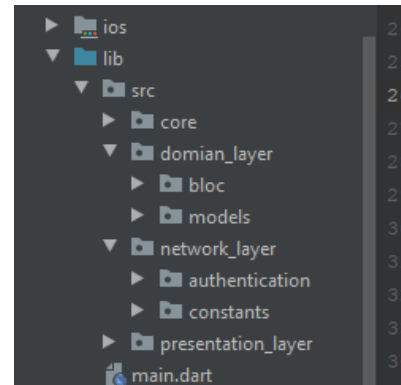

Fig: Layering and Folder Structure in Client Side App

any changes are made in the firebase collections. The Firebase Messaging Token is used to identify each user to their logged devices.

The architecture given below shows a Flutter client Mobile Application, an Admin Panel, the Firebase Services, Firebase Collections, Firebase Storage and Cloud Functions. Various functionalities are defined in the client device namely, Login, Profile Update, Assignments, Mobile ID. Whenever the user clicks on his profile view widget then a request will be fired by the client to the firebase services. This request will then fetch user profile details from the database storage i.e., Collections if the profile also contains a profile image, then the profile image will also be fetched from the storage bucket. All this will be returned in the response. Then the client will update and render the ui accordingly.

Similarly, if the admin panel posts an announcement, then firebae services are called accordingly and additions are made in the collections. This also fires an additional request to GCP where cloud triggers are written in a way such that when an announcements data is added then push notifications to all the users will be sent. The messaging token is used to identify each user to its corresponding logged in devices.

3.2 Client-Side Application Layering

As more and more functionalities are added to the application the code becomes more complex and fattier. It is very important that we layer the code such that boiler plate is avoided and also the structure is maintained by all developers. This also enables changes and faults to be identifiable

phase of brainstorming and drafting all the possible features that will be related to the requirements in the system. The design phase includes database, system and user flow designs that detail out how the above gathered requirements will be implemented. Wireframing of the application is done to see how and which screen will cater to what functionalities. These wireframes are then converted to User Interface adding colours and interactions to the wireframe designs. As more than one developer works on the application some guidelines are to be set such that they are followed by all the developers and clean and maintainable code is drafted. Version control is setup for sharing the code and combining each other's work. Also to keep track of versions and when new functionalities were implemented. Pseudo code is drafted so that logic and design of the system can be matched overall. Once all these steps are complete then the actual development starts.

Fig: UI/UX development Phase with dummy Data

As currently the system is not built to support video classes, we will extend it to add video conferencing options using sockets. Also, then face recognition can be added there for automated attendance tracking.

6. EXPERIMENTAL RESULTS

Table 1 Unit Test Case 1

S1 # Test Case	UTC-*1
Name of Test	Registration
Expected Result	Admin Panel should be able to create a user profile for a student, generate its unique id.
Actual output	Same as expected.
Remarks	Successful

Table 2 Unit Test Case 2

S2 # Test Case	UTC-*2
Name of Test	Student login.
Expected Result	Student should be able to login and see the appropriate dashboard.
Actual output	Same as expected.
Remarks	Successful

Table 3 Unit Test Case 3

S3 # Test Case	UTC-*3
Name of Test	Student to retrieve course Info
Expected Result	Student should see the details of the course he/she has enrolled in this semester
Actual output	Same as expected.
Remarks	Successful

Table 4 Unit Test Case 4

S4 # Test Case	UTC-*4
Name of Test	Time Table
Expected Result	Student Should be able to see only his time table

4. LIMITATIONS

The major limitation of this system is that the backend services user here are Firebase and Google Cloud Functions. These services are free to use until a certain user and bandwidth limit. After which these become very expensive. Hence, while in production and distribution the backend service needs to be shifted to a better alternative. Currently most educational institutes are using free applications like Slack and Google Classroom, this system is not capable of hosting video conference currently thus video-based classes which will be conducted on other platforms need to be added to the student's timetable manually. There is no API or implementation to automate this process.

5. FUTURE WORK

We will extend the system by creating a custom website for the admin panel. Currently the admin panel in the firebase console from where new registrations can be done manually.
Plagerism checkers can be added in the assignment upload section.

Actual output	Same as expected.
Remarks	Successful

Table 5 Unit Test Case 5

S5 # Test Case	UTC-*5
Name of Test	Post Announcements
Expected Result	Admin User should be able to add notices and announcements from the admin panel.
Actual output	Same as expected.
Remarks	Successful

Table 6 Unit Test Case 6

S6 # Test Case	UTC-*6
Name of Test	Push Notifications
Expected Result	When admin adds a new notice, it should be notified to all users
Actual output	Sometimes the user receives the notification and sometimes not
Remarks	Unsuccessful

Table 7 Unit Test Case 7

S7 # Test Case	UTC-*7
Name of Test	Fetch Mobile ID
Expected Result	Student should be able to fetch his/ her mobile Id from cloud
Actual output	Same as expected.
Remarks	Successful

5. CONCLUSIONS

After the COVID19 pandemic the online industry has taken a boom advancement. Various platforms have come up with various solutions but a required solution for our education platforms to manage resources that is specific to institute requirements are not sufficient. Thus, this application will solve this problem and provide a single platform for all data entry and dashboard management.

8. REFERENCES

[1] S.D. Drake, "Embracing Next-Generation Mobile Platforms to Solve Business Problems", a Sybase White Paper, Oct 2008. http://www.sybase.com/detail?id=1060699. Accessed 7/4/2009.

[2] A Comparision of Performance and Looks Between Flutter and Native Applications(2020)
Authors: Matilda Olsson, June 2020
[http://www.diva-portal.org/smash/get/diva2:1442804/FULLTEXT01.pdf]

[3] Mobile Application Development with Flutter (2020)
Authors: lähettänyt Henri Mäkelä päiväys: [https://blog.boogiesoftware.com/2019/10/mobile-application-development-with.html]