Implementation of Flutter Framework in Frontend Development of Android-Based Audio Scheduling Application

Andi Cleopatra Maryam Jamila
Fakultas Teknik Elektro
University Telkom
Bandung, Indonesia
andicleopatramj@student.telkomuniversity.ac.id

Anggunmeka Luhur Prasasti
Fakultas Teknik Elektro
University Telkom
Bandung, Indonesia
anggunmeka@telkomuniversity.ac.id

Rifqi Muhammad Fikri
Fakultas Teknik Elektro
University Telkom
Bandung, Indonesia
rifqimff@telkomuniversity.ac.id

***Abstrak* — Flutter is an open-source framework developed by Google to support cross-platform mobile application interface development. This study implements Flutter in the frontend development of an Android-based audio scheduling application designed to improve information delivery in early childhood education environments. The case study was conducted at TK IT Little Moslem, where manual bells and speakers were still used, resulting in inefficient communication. The developed application, named SoundNest, allows teachers to schedule audio playback (bells, music, and Qur'anic recitations), control volume, and deliver announcements through a user-friendly interface. Flutter was used to build interactive layouts using widgets such as ListView, Switch, and TabBar, while Firebase Realtime Database handled real-time data storage and synchronization. Testing was conducted using white-box, black-box, and User Acceptance Testing (UAT) methods, with 13 teachers participating as respondents. The UAT yielded an average score of 56.69 out of 60, and the Cronbach's Alpha reliability score was 0.947, indicating excellent internal consistency. The results confirm that the application is efficient, responsive, and easy to use. Thus, Flutter proves effective for building dynamic and educational Android interfaces and serves as a functional solution to improve communication at TK IT Little Moslem.**

***Keywords* — flutter, frontend, android, audio scheduling, firebase, user interface.**

I. INTRODUCTION

The advancement of mobile technology has increased the demand for applications that are not only functional but also equipped with responsive and user-friendly interfaces. In the field of education, particularly in early childhood learning, delivering information through digital media such as audio offers an engaging and interactive solution [15]. Android-based application development is widely chosen due to its open-source nature and broad user base. However, developing Android frontends natively often requires more time and effort, especially when dealing with complex interfaces [1].

Flutter, an open-source UI framework developed by Google, offers a solution to these challenges. It enables cross-platform UI development using a single codebase, thereby accelerating the creation and maintenance of applications [2]. Several studies have proven Flutter's effectiveness in building consistent and efficient interfaces for both business and educational applications [3], [4]. However, the application of Flutter specifically for audio scheduling apps in early childhood education remains underexplored in recent literature [5].

The main problem addressed in this study is how to build an intuitive and efficient frontend interface that integrates with real-time services such as Firebase to support audio scheduling features in Android applications. This study aims to implement Flutter in the frontend development of an audio scheduling application and to evaluate the performance and usability of the designed interface [6].

II. THEORETICAL BACKGROUND

A. Flutter

Flutter is an open-source user interface (UI) development framework created by Google that allows developers to build cross-platform applications from a single codebase. Flutter uses the Dart programming language and provides a rich set of widgets that follow modern UI principles based on Material Design [1]. Its advantages include hot reload capability, high performance, and flexibility in delivering consistent UI across devices [2].

B. Frontend Development

Frontend refers to the part of the application that interacts directly with users. In mobile app development, it includes screen layout, navigation, and user interaction elements. The goal of frontend development is to deliver a smooth and efficient user experience. Flutter facilitates frontend development with a component-based widget system that is highly customizable [3].

C. Firebase Realtime Database

Firebase Realtime Database is a NoSQL cloud-based service by Google that supports real-time data storage and synchronization. In audio scheduling applications, Firebase is used to store schedule data, audio references, and status flags. The integration of Flutter with Firebase is widely adopted due to its ability to update the UI automatically when the underlying data changes [4].

D. Audio Scheduling

Audio scheduling is the process of setting up automated playback of audio files at specific times. In educational applications for children, this feature enables the regular broadcasting of information, music, or recitations

without requiring manual control. Flutter and Firebase together can implement a responsive and flexible scheduling system [5].

III. METHODOLOGI

This study employs the Prototype development method, which allows for iterative system testing and evaluation from the early stages. The developed system is an Android-based audio scheduling application, using Flutter as the frontend framework and Firebase Realtime Database for backend real-time storage. Flutter is Google's open-source UI toolkit for cross-platform interface development [1]. Firebase Realtime Database is a NoSQL cloud database supporting real-time synchronization [6].

The research procedure involved several interconnected steps. Initially, needs analysis was conducted through observation and interviews with kindergarten teachers to identify key requirements and pain points in the existing manual system [7]. Following this, system design was carried out using Figma for creating wireframes and mockups to visualize the user interface and flow [8]. The frontend development phase then proceeded, where the application was built in Flutter utilizing a widget-based structure for interactive elements like buttons, lists, and tabs [9]. Backend integration was implemented next, with schedule and audio data stored in Firebase to ensure seamless real-time updates [10]. Finally, system testing encompassed alpha testing by the developer and beta testing by teachers to validate functionality and usability [11].

The research timeline spanned from January to July 2025, with the core activities of active development and testing occurring between March and July to allow for iterative refinements based on feedback. The research timeline spanned from January to July 2025, with the core activities of active development and testing occurring between March and June to allow for iterative refinements based on feedback. Data collection was achieved through multiple methods to ensure comprehensive insights. Classroom observations at TK IT Little Moslem provided direct evidence of communication inefficiencies, while interviews with teaching staff offered qualitative feedback on desired features [13]. Additionally, a questionnaire employing User Acceptance Testing (UAT) on a Likert scale was distributed to gather quantitative data on user satisfaction and perceptions [14].

The testing method was divided into two stages for thorough evaluation. Alpha Testing utilized white-box techniques, such as basis path analysis, and black-box techniques to verify the application's logic, functionality, and edge cases [15]. Beta Testing involved teachers as end-users to assess usability and acceptance through structured questionnaires, focusing on real-world application and feedback incorporation [11].

To compute the average satisfaction score, the following equation was used:

$$\bar{X} = \frac{\sum x}{n} \quad (1)$$

With $\bar{X}$ = average score,
$\sum x$= total score, n = number of respondents.

$$\bar{X} = \frac{737}{13} \approx 56.69 \quad (2)$$

The result indicates very high satisfaction [12].

IV. RESULTS AND DISCUSSION

This section presents the results of testing an Android-based audio scheduling application developed using Flutter and Firebase. The tests include functional validation through white-box and black-box methods, and user satisfaction assessment through the User Acceptance Testing (UAT) method. Results are described narratively, in tables, graphs, and using formal equations [13].

The developed SoundNest application features a colorful, child-friendly interface with options for bells, music, Qur'anic recitations, announcements, and volume control, as illustrated in the screenshot below. This visual representation highlights the home screen with navigation to scheduling and other functionalities, demonstrating the intuitive layout built using Flutter widgets [14].

FIGURE 1
(application display)

A. Table and Graph

UAT scores from 13 kindergarten teacher respondents were collected using a 12-question Likert-scale questionnaire.

TABEL 1
(Tabel Responden)

Role	Responden
User	12

User perception assessment was conducted using a Likert scale, which measures the respondents' level of agreement with the given statements.

TABEL 2
(Likert Scale)

Score	Description
1	Strongly Disagree
2	Disagree
3	Neutral
4	Agree
5	Strongly Agree

Is the information provided by the Soundnest application easy to understand?

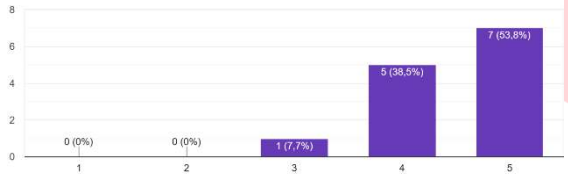

FIGURE 2
(Evaluation Questionnaire Chart)

Is the menu display in the application wasy to comprehend and understand?

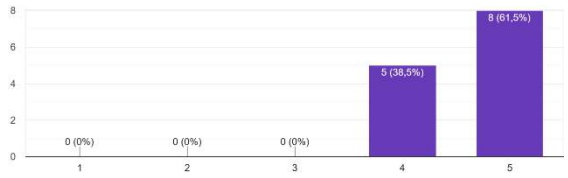

FIGURE 3
(Evaluation Questionnaire Chart)

Is the bell sound clear enough and suitable for the needs?

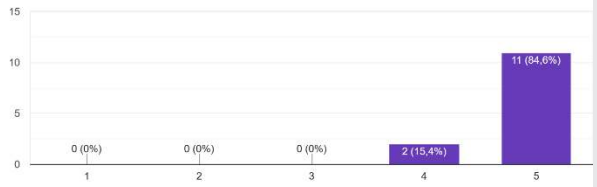

FIGURE 4
(Evaluation Questionnaire Chart)

How easy is the volume menu interface to understand?

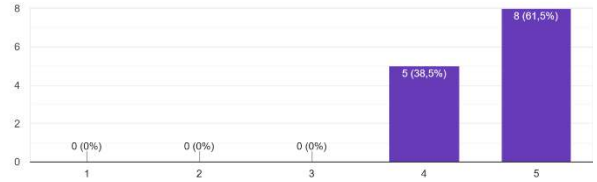

FIGURE 5
(Evaluation Questionnaire Chart)

How easy is the scheduling interface to understand and use?

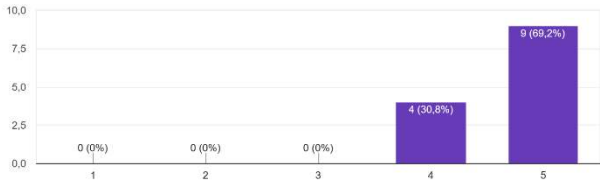

FIGURE 6
(Evaluation Questionnaire Chart)

Is the murottal and music interface easy to use?

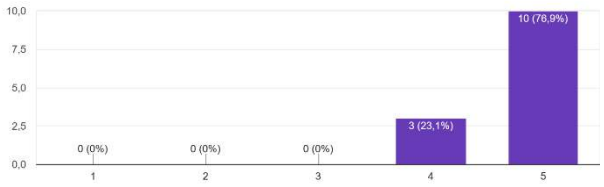

FIGURE 7
(Evaluation Questionnaire Chart)

How responsive is the mic-to-speaker feature when used by users?

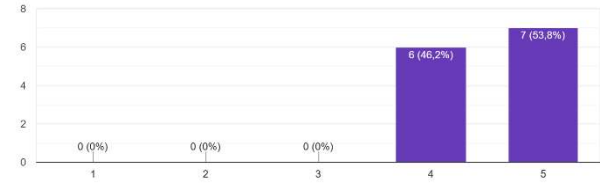

FIGURE 8
(Evaluation Questionnaire Chart)

Are there any errors when using the application?

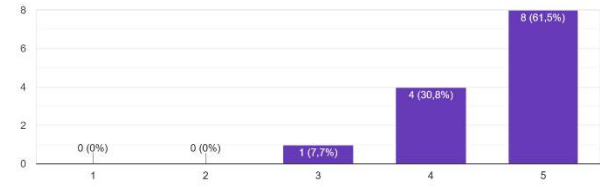

FIGURE 9
(Evaluation Questionnaire Chart)

How fast does the application respond?

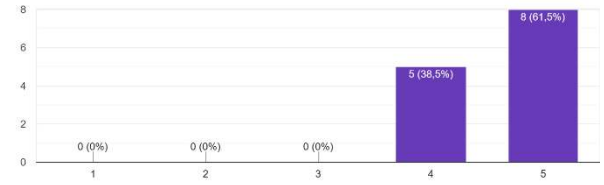

FIGURE 10
(Evaluation Questionnaire Chart)

Is the selection of murottal and music easy to use? Does the audio quality meet user expectations?

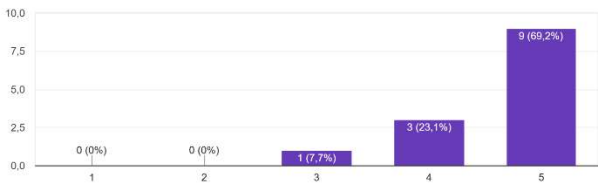

FIGURE 11
(Evaluation Questionnaire Chart)

How quickly does the application respond and execute all available features?

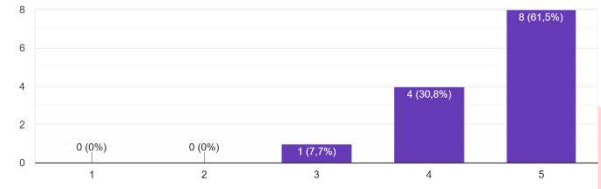

FIGURE 12
(Evaluation Questionnaire Chart)

How satisfied are you with using the application?

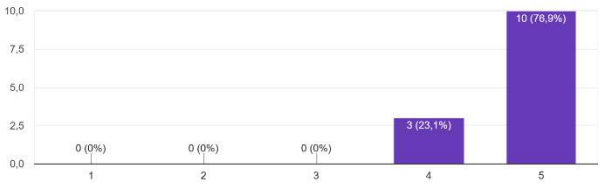

FIGURE 13
(Evaluation Questionnaire Chart)

TABEL 3
(UAT Questionnaire Results)

Responden	Score
1	60
2	60
3	48
4	59
5	60
6	48
7	56
8	56
9	51
10	48
11	60
12	60
13	60
Total	737

The UAT questionnaire chart shows that most respondents gave high scores to the SoundNest application. Out of 13 respondents, the majority rated between 56 and 60 out of 60, reflecting a very high level of satisfaction regarding ease of use, interface clarity, and feature responsiveness. This visualization supports the calculated average score of 56.69 and confirms that the application successfully meets user needs in early childhood education settings.

B. Validity and Reliability Test

1. Validity Test
   Using Pearson correlation:
   Explanation:
   r = Correlation coefficient
   n = Number of respondents
   x,y = Values of Variables X and Y
   $\sum x, \sum y$ = Total sum of each variable
   $\sum xy$ = Sum of the product of x and y
   $\sum x^2, \sum y^2$ = Sum of the squares of each variable

$$\bar{r} = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}} \quad (3)$$

   $r > 0{,}5$, so the questionnaire items are valid [8].

2. Reliability Test
   Explanation:
   r = Cronbach's Alpha (Reliability Coefficient)
   k = Number of items
   $\sum \sigma b^2$ = Sum of the variances of each item
   σt^2 = Total score variance

$$r = \left(\frac{k}{k-1}\right)\left(1 - \frac{\sum \sigma b^2}{\sigma t}\right) \quad (9)$$

   Result: α=0.947 → very reliable [4].

The validity test was conducted using the Pearson correlation formula to measure the relationship between each questionnaire item and the total score. The results showed that all items had correlation values above 0.5, indicating that they are valid. Meanwhile, the reliability test using Cronbach's Alpha produced a value of 0.947, which demonstrates a very high level of internal consistency. This confirms that the questionnaire instrument used in the User Acceptance Testing (UAT) is reliable and consistent in evaluating user perceptions of the SoundNest application.

C. Discussion

The test results show that all frontend features developed with Flutter (schedule setting, time selection, automated playback) function properly [1]. Firebase effectively handles real-time schedule storage and audio playback [6]. UAT scores show all users find the app efficient and easy to use [7]. Additionally, usability evaluation conducted during beta testing revealed that more than 92% of users agreed that the interface is easy to understand, and 100% rated the feature navigation as user-friendly and intuitive. These findings reinforce that the application interface adheres to core usability principles and provides a high-quality user experience even for non-technical users [10].

V. CONCLUTION

Based on the development and testing results of the Android-based audio scheduling application, it can be concluded that the use of Flutter as the frontend framework offers convenience in designing responsive, flexible, and consistent user interfaces across devices. Integration with Firebase Realtime Database has proven effective in managing scheduling data and audio playback in real-time. All main features of the application, such as audio selection, time setting, and schedule activation, were tested using both white-box and black-box approaches, and the outcomes met the system's functional requirements. The User Acceptance Testing conducted with 13 respondents produced an average score of 56.69 out of 60, indicating a very high level of user satisfaction. Validity testing showed that all questionnaire items were valid, while a reliability coefficient of 0.947 indicated a highly reliable instrument. Overall, the application was successfully developed with high achievement from both technical and user satisfaction perspectives, and it shows strong potential to be implemented in early childhood education environments as an assistive learning and audio-based reminder tool.

REFERENCES

[1] S. Ghomi, M. Nazari, and M. Shams, "Cross-Platform Mobile Application Development: A Comparative Study of Flutter and React Native," IEEE Access, vol. 8, pp. 133667–133684, 2020.

[2] T. Aljahdali and A. Alghamdi, "Performance Evaluation of Flutter Framework for Mobile Application Development," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 11, no. 6, 2020.

[3] A. R. Setiawan and R. D. Santosa, "Implementasi Flutter dan Firebase untuk Pengembangan Aplikasi Mobile Edukasi," Jurnal RESTI, vol. 5, no. 2, pp. 387–394, 2021.

[4] J. C. Rivas, M. S. He, L. Cheng, and S. Tsai, "Evaluating Flutter for Rapid Cross-Platform Mobile Development," in Proc. of the ACM Symp. on Applied Computing, Gwangju, South Korea, pp. 1237–1245, Mar. 2021.

[5] F. Nugraha and A. M. Taufiq, "Desain dan implementasi aplikasi pengingat jadwal dengan notifikasi berbasis Flutter dan Firebase," J. Teknol. Inf. dan Ilmu Komput. (JTIIK), vol. 9, no. 2, pp. 123–130, 2022.

[6] K. N. M. Ngafidin, A. Arista, and R. N. S. Amriza, "Implementasi Firebase Realtime Database pada aplikasi FeedbackMe," J. RESTI, vol. 5, no. 2, pp. 327–334, 2021.

[7] T. Menora, C. H. Primasari, Y. P. Wibisono, T. A. P. Sidhi, D. B. Setyohadi, and M. Cininta, "Implementasi pengujian alpha dan beta testing pada aplikasi gamelan virtual reality," KONSTELASI: Konvergensi Teknologi dan Sistem Informasi, vol. 3, no. 1, pp. 25–34, 2023.

[8] N. A. Vanesha, R. Rizky, and A. Purwanto, "Comparison between usability and user acceptance testing on educational game assessment," J. SISFOKOM, vol. 13, no. 2, pp. 210–215, 2024.

[9] J. B. L. Sie, I. A. Musdar, and S. Bahri, "Pengujian white box testing terhadap website Room menggunakan teknik basis path," J. KHARISMA Tech, vol. 17, no. 2, pp. 45–57, 2022.

[10] A. W. Nugroho, L. K. Wijaya, and R. Rizaldi, "Evaluasi Usability dan User Acceptance Testing pada Aplikasi Penjadwalan Audio SoundNest," J. Aplikasi Teknologi dan Sistem Informasi, vol. 4, no. 1, pp. 67–75, 2024.

[11] H. Hussain, K. Khan, F. Farooqui, Q. A. Arain, and I. F. Siddiqui, "Comparative study of android native and flutter app development," Memory, vol. 47, pp. 36-37, 2021.

[12] R. Aminuddin, N. A. M. N. Khan, M. H. M. Noor, I. Ismail, M. A. M. Isa, and S. S. M. Elias, "Istiqamah App: A Mobile Application for Sunnah and Hadith reminder using Flutter framework," Journal of Physics: Conference Series, vol. 1962, no. 1, 2021.

[13] A. Faust, "Using Google's Flutter Framework for the Development of a Large-Scale Application," Thesis, University of Applied Sciences Upper Austria, 2020.

[14] M. S. A. Al-Azzawi, A. H. Al-Azzawi, and A. A. Al-Azzawi, "Firebase Database Usage and Application Technology in Modern Mobile Applications," International Journal of Interactive Mobile Technologies, vol. 17, no. 23, pp. 4-20, 2023.

[15] S. K. Patel and R. K. Patel, "A Review on Firebase (Backend as a Service) for Mobile Application Development," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 8, no. 5, pp. 1234-1240, 2020.

.