Dynamic Contact App: The Case Study of JNEC Contact

Rudra Dungyel[1], Dendup Tshering[2], Younten Tshering[3*], Tashi Wangchuk[4], and Tashi Yangchen[5]

[1-5]*Department of Information Technology, Jigme Namgyel Engineering College, Royal University of Bhutan*
[*]*Corresponding author: yountentshering.jnec@rub.edu.bt*

Published: February 2025

Abstract

Even though efficient communication is crucial in todays workplace, many businesses struggle with inconsistent contact management platforms and disjointed data sources. Inefficiencies are frequently caused by this fragmentation since employees find it difficult to obtain available and correct contact information when needed. Furthermore, juggling several communication channels can cause miscommunication and delays, which impedes teamwork and productivity. This highlights the necessity for a dedicated contact management application, the Contact App. Using Flutter for cross-platform development and Firebase for real-time database and authentication, the application is dynamic and compatible with Android and iOS devices. This application simplifies the process of managing contacts, allowing users to store, update, and access contact details with ease, and also increases overall workplace productivity by making this application a complete and effective solution for contemporary corporate communication. In addition, the application provides an interactive interface for users to call, email, and send messages.

Keywords— Application, dynamic, flutter, firebase

1 Introduction

In an era where digital communication is pervasive, the need for a centralized and efficient system to manage contact information has become increasingly clear. Traditional methods of maintaining contacts, such as physical address books or scattered digital files, often result in disorganization, inefficiency, and missed opportunities. Whether it is for personal connections or professional networking, managing contact information efficiently is key to staying organized and building meaningful relationships. As per [1], the JNEC Contact app was deployed by giving the Android Application Package (APK) file or setup file to all Android phone users. Although Android phone users found the contact application user-friendly and convenient in their daily lives, it was available only for Android devices. The iPhone Operating System (iOS) users preferred the application to be compatible with their iOS system. Similarly, the feedback given for the improvement of the app was to make the application dynamic so that the contact details can be modified when required and some users with technical backgrounds suggested integrating the database. Hence, this paper became the main pillar of reason to embark on this project to develop it into a dynamic Contact App. Recognizing the challenges, this project embarked on the development of the Contact App, a modern solution tailored to meet the evolving needs of contact organization and accessibility. Using Flutter for cross-platform development and Firebase for real-time database and authentication, the application is made dynamic and compatible with both Android and iOS devices.

2 Literature Review

As per [2], mobile app development is made possible using different techniques and methodologies that add essence to the application. Its a long process that cannot be achieved all at once. Iterative development is the way of breaking down the development of large application software into small chunks so that all the features are brought out to the users with a lot more efficiency. The iterative approach to mobile app development provides the ease of getting it done in small steps. By repeatedly focusing on the entire process of app development, it helps in completing the application development with more efficiency. Therefore, the same method was implemented as mentioned above to design and develop the Contact App. According to [3], building an app in Flutter can be smooth, seamless, and cost-effective. With Flutter, costs can be saved by cutting down the amount of coding needed. This helps reduce expenses as opposed to creating apps, for two platforms, which would require writing the code twice. Besides the costs, apps built in Flutter perform similarly to native apps. Flutter has some technology advantages such as built-in widgets and command line tools [4]. It relies less on third-party tools and has great testing, CI/CD, and maintenance tools which can make it easy to learn and quickly become productive. Clearly, with a growing community and Google, its creator, shifting more focus toward it, the prospects will be better. Therefore, the Flutter framework was chosen for cross-platform development for this project. In the report [5], Firebase is considered as a web application platform. It helps developers build high-quality apps. Firebase is NoSQL based. It stores the data in JavaScript Object Notation (JSON) format which doesnt use a query for inserting, updating, deleting, or adding data to it. Firebase provides the backend and also the database to store data of this project.

3 Methodology

The Contact App was developed with an iterative approach where the application development process is divided into small, manageable chunks called iterations. This process is repeated until the final product meets the desired requirements for the deployment of the application. This application development included the following phases as shown in Figure 1.

3.1 Planning

During the planning phase, all the requirements for the app development were collected before getting started with the design and the rest of the phases. After identifying the requirements, they were segregated into functional and non-functional requirements and analyzed to meet the system's needs and ensure user satisfaction. Brainstorming and analysis, review of existing application, and documentation were conducted for the collection of requirements.

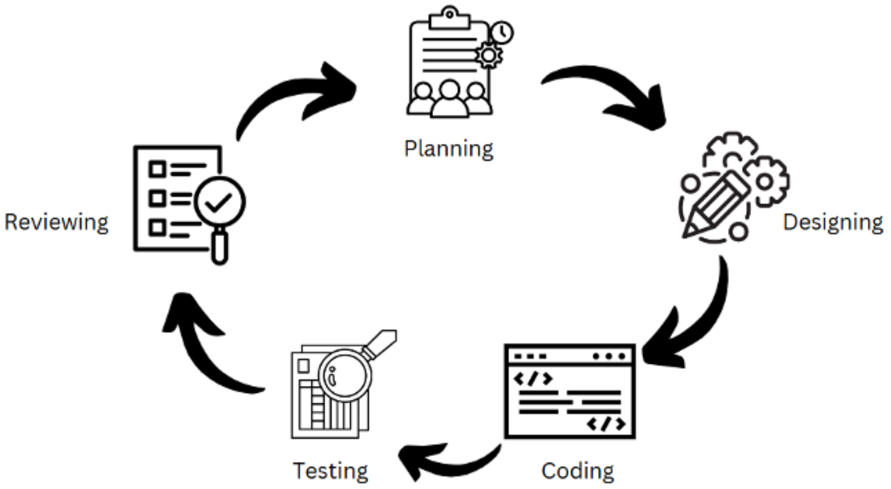


Figure 1: Iterative Process Model

3.2 Designing

Draw.io (diagrams.net), a free, open-source, and versatile diagramming application was used for creating the use case diagram to describe the functional requirements of the system. This phase facilitated the understanding and documentation of system behavior from the user's perspective.

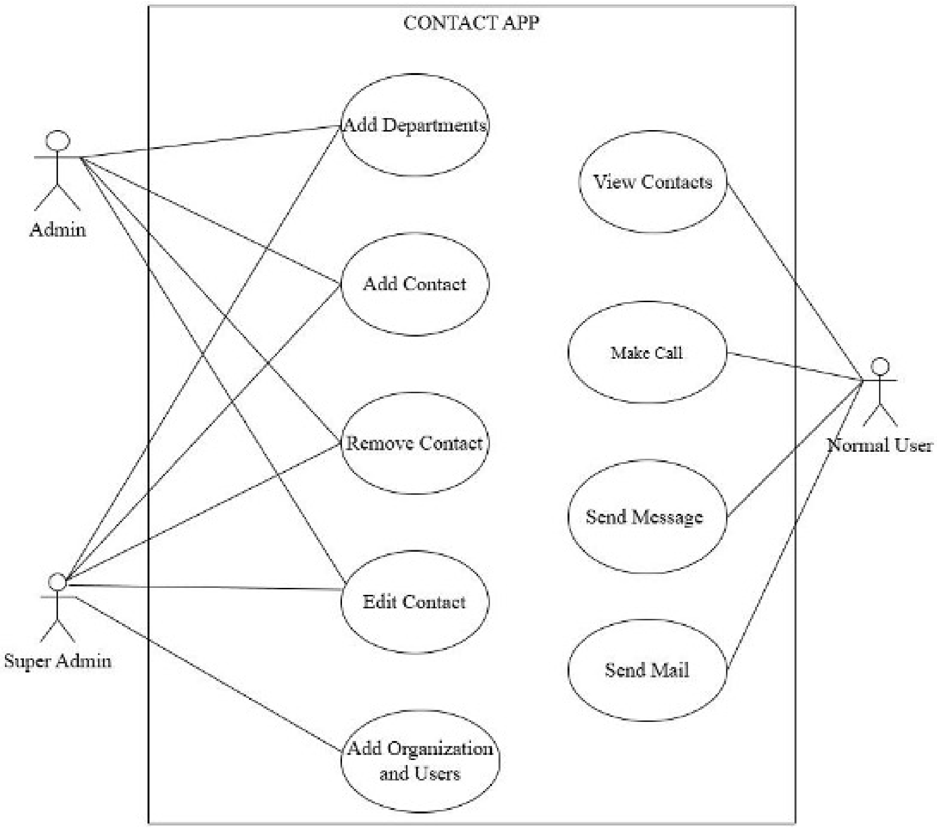


Figure 2: Use case Diagram

3.3 Coding

With the completion of the design, coding was initiated with the pair programmer concept using Flutter and Dart as the programming language in the Visual Studio Code. In this approach, one of the developers acted as the driver focused on coding, and the other as the navigator focused on fixing the error simultaneously. The roles were periodically swapped to ensure both programmers remained actively engaged and shared a deep understanding of the code. In the due course of app development, functional testing was carried out to ensure that it was as per the design.

3.4 Testing

Unit testing, white-box testing, and integration testing were a concern during the development and implementation of the application. For the successful operation of the application, vigorous unit testing was done to verify and validate its functionality. Once coding was completed and the final version was ready, system testing of the whole app was conducted on various Android versions using emulators.

3.5 Reviewing

Constant feedback and insightful reviews of the testing outcomes were received from the project guide throughout the project. This process was an essential part of the iterative approach's feedback loop, promoting agility, adaptability, and responsiveness to changing needs throughout the software development lifecycle. Each cycle or sprint was aimed at incrementally improving the app by adding new features or refining the existing ones. The insights and feedback from each review phase were used for the planning of the next phase, ensuring continuous improvement and alignment with user needs.

4 Result

Users are successfully authenticated and logged into the app with an email and password. For the app, users are divided into three categories based on the roles that are assigned to the authorized email as shown in Figure 3. The following are the different types of users and the permissions assigned.

- Normal user can view and have access to contact information of their organization. They can also make calls, and send text messages and email.
- Admin has authority to modify contact information in their organization.
- Super admin has complete authority over any organization and the privilege to create new organizations and users in the app.

The application was installed, launched, and used on an Android device. To run the application on an Android device, the developer option is enabled on the device. This allows the application and also debugging of the application. After the app launches, it opens the login page interface for the user to log in as shown in Figure 4. The users are only able to log in to the app through the authenticated email and password that are shared by the Super Admin of the app or the Admin of the organization. When the user logs in as the Super Admin, the app will open the Super Admin interface. Figure 5 shows the Super Admin page to select an organization from the organization lists and modify the information embedded in it. It also provides an interface to create an organization and the users (Admin and Normal User) for that organization.

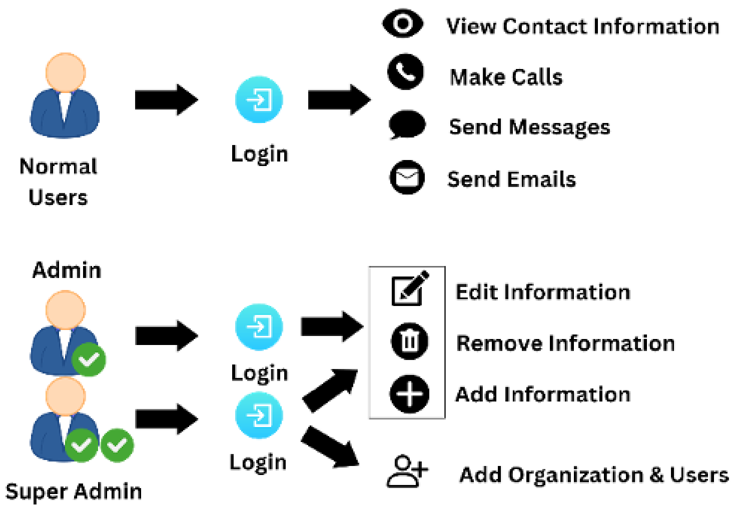
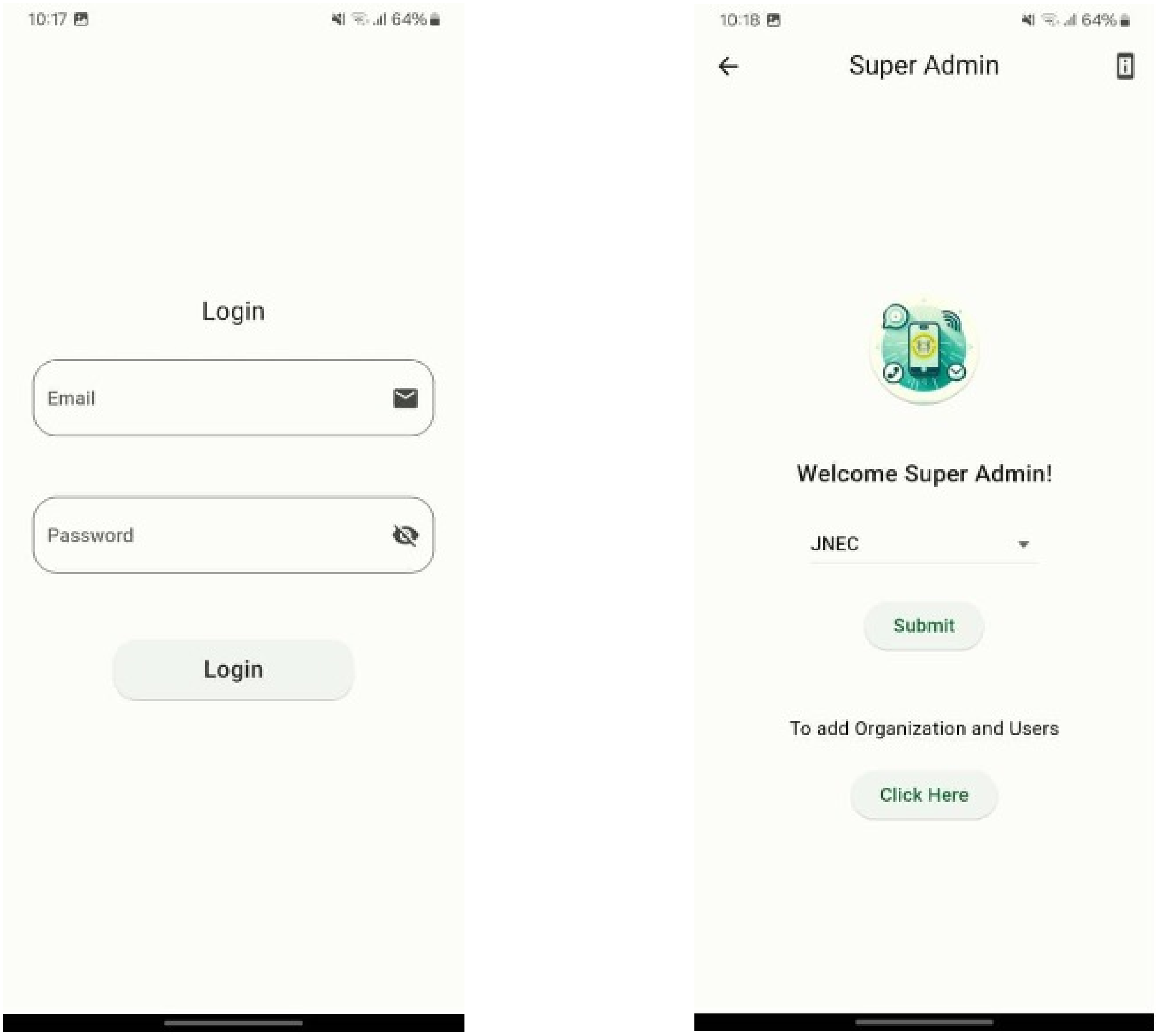


Figure 3: System Functionality



Welcome Super Admin!

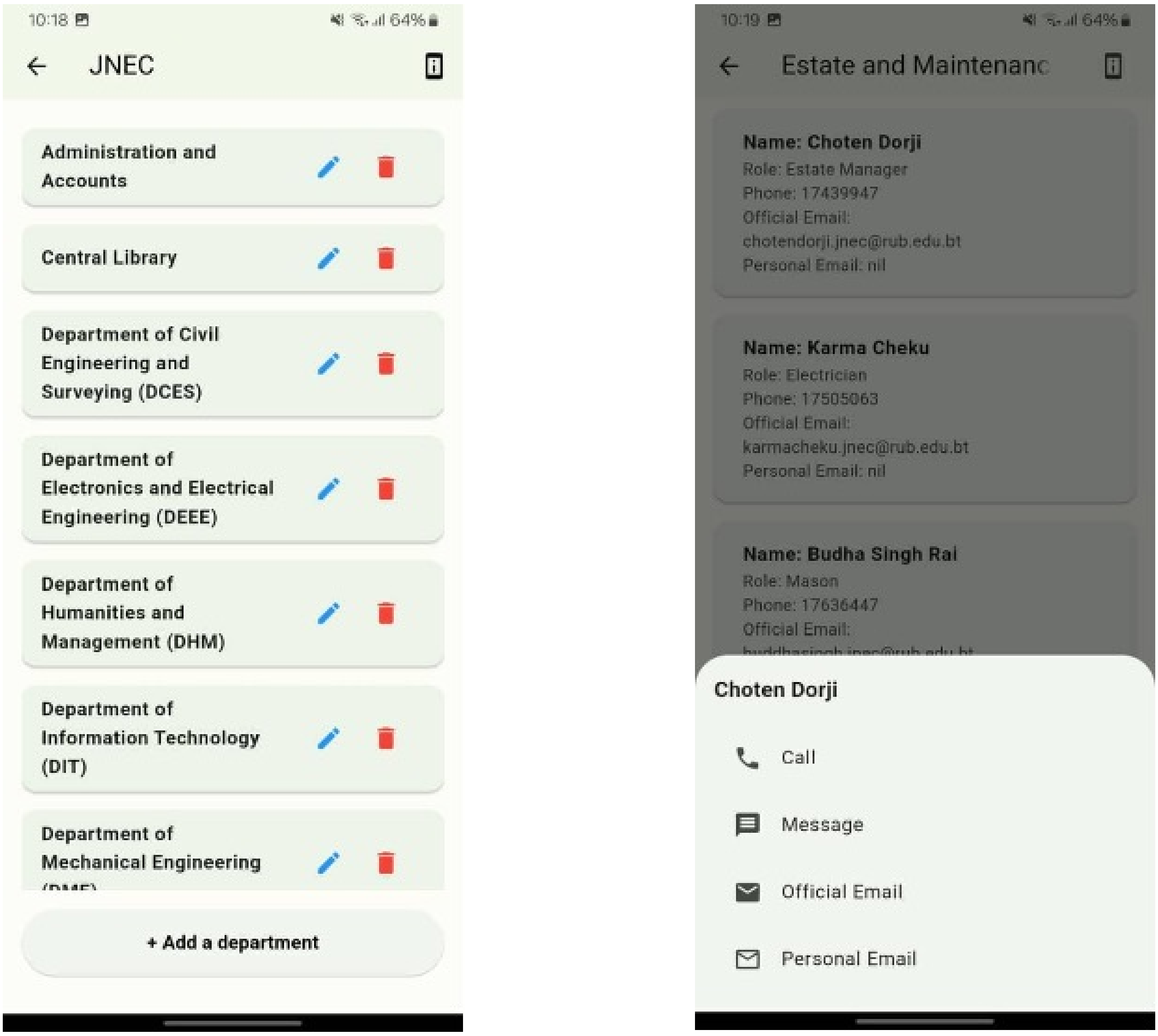
To add Organization and Users

(a)

(b)

Figure 4: (a) Login Page, and (b) Super Admin Page

If the user logs in as Super Admin or Admin, the Department Page provides a comprehensive set of features to manage the department information as shown in Figure 6. The user can perform add, edit, and delete operations on the department list. When clicking on any department from the department list, the app will navigate to the Staff Page. If the user logged in is Super Admin or Admin, the Staff Page provides a comprehensive set of features to manage the contact information details. The user can perform add, edit, and delete operations to the contact details. However, if the user logged in is a Normal User, the Department Page will provide just the list of departments of the organization and will not be able to perform add, edit, and delete operations to the department list like Super Admin or Admin. Upon clicking on any department from the department list, the app will navigate to the Staff Page. If the user logged in is a Normal User, the Staff Page will provide just the list of contact information details of that department. The user will not be able to perform add, edit, and delete operations to the contact details like Super Admin or Admin. When the user clicks on any contact details, the app provides the user with an interface option to call, send messages, or send emails as shown in Figure 7. This is an important functional feature of this application that provides an interactive interface for users to directly call, email, and send messages.



Administration and Accounts

Central Library

Department of Civil Engineering and Surveying (DCES)

Department of Electronics and Electrical Engineering (DEEE)

Department of Humanities and Management (DHM)

Department of Information Technology (DIT)

Department of Mechanical Engineering

Name: Choten Dorji
Role: Estate Manager
Phone: 17439947
Official Email:
chotendorji.jnec@rub.edu.bt
Personal Email: nil

Name: Karma Cheku
Role: Electrician
Phone: 17505063
Official Email:
karmacheku.jnec@rub.edu.bt
Personal Email: nil

Name: Budha Singh Rai
Role: Mason
Phone: 17636447
Official Email:

Choten Dorji

(a) (b)

Figure 5: (a) Department Page, and (b) Contact Options Interface

5 Discussion

The Contact App was successfully developed as a dynamic and user-responsive application, replacing the existing static application and eliminating the conflicts of accessing outdated data and limited compatibility features to enhance overall user satisfaction. It provides an intuitive and user-friendly interface that enables users to navigate the application effortlessly, providing easy access to contact information and a comprehensive suite of features to simplify contact management tasks. This robust and dynamic application, developed to be compatible with both Android and iOS devices replaces the existing Android-based static application. However, the application was not tested with real users and was not deployed in the Play Store and App Store.

6 Conclusion

The "Contact App" marks a pivotal shift in organizational communication, ensuring efficiency and inclusivity across multiple operating systems and addressing issues with disjointed contact management. Developed iteratively, it adapts to stakeholders' evolving needs, refining its user-centric features through continuous analysis, design, execution, and assessment. The application's comprehensive features cater to diverse user needs, from robust user identification to intuitive contact management and communication tools. Super Admin and Admin roles facilitate organizational management by allowing contact additions, updates, and removals, ensuring scalability and adaptability. By emphasizing on non-functional requirements such as compliance, user-friendliness, cross-platform compatibility, security, performance, and reliability, the application demonstrates a commitment to high-quality mobile development. With stringent data protection, speed optimization, and privacy measures, it sets a new standard in mobile applications. In the dynamic digital market, the application exemplifies innovation, enabling businesses to implement advanced communication strategies and enhance productivity. Its dynamic nature and cross-platform accessibility make it a valuable tool for improving workplace efficiency, collaboration, and communication.

References

[1] Y. Tshering, S. Dorji, and T. Wamgmo, "User Friendly Android-based Contact Application: Case of Accessible Contact in Educational Organization," *iJRASET*, vol. 10, no. 9, Sep. 2022. [Online]. Available: `https://www.ijraset.com/best-journal/user-friendly-androidbased-\contact-application-case-of-accessible-contact-in-educational-organization`

[2] V. Shah, "Top reasons to opt for iterative approach to mobile app development," *dev*, Apr. 28, 2018. [Online]. Available: `https://dev.to/vaibhavshah/top-reasons-to-opt-for-iterative-approach-to-mobile-app-development-173d`

[3] B. Szypelow, "Why is Flutter a good choice for cross-platform projects?," *Netguru*, Dec. 6, 2023. [Online]. Available: `https://www.netguru.com/blog/flutter-for-cross-platform-app-development`

[4] D. Erickson, "Cross Platform Development Technology Comparison: React Native vs. Flutter," *SCREAMINGBOX*, Mar. 27, 2024. [Online]. Available: `https://www.screamingbox.net/blog/cross-platform-development-technology-comparison-react-native-vs-flutter`

[5] C. Khawas and P. Shah, "Application of Firebase in Android App Development-A Study," *International Journal of Computer Applications*, vol. 179, no. 46, Jun. 2018. [Online].

Available: https://www.ijcaonline.org/archives/volume179/number46/khawas-2018-ijca-917200.pdf

[6] A. Jaiswal, "Contact Management System," *Galgotias University.* [Online]. Available: http://103.47.12.35/bitstream/handle/1/1603/contect%20management%20system%28report%29%20-%20Abhishek%20Jaiswal.pdf?sequence=1&isAllowed=y

[7] A. Krishnan, "Flutter vs React Native in 2023: A Detailed Comparison," *hackr*, Jan. 4, 2023. [Online]. Available: https://hackr.io/blog/react-native-vs-flutter

[8] HeyFlutter, "Flutter Firebase Setup using Flutter Fire CLI [2023] Easy and Fastest Way!," *YouTube*, 2023. [Online]. Available: https://www.youtube.com/watch?v=G-mbqiE87Lw

[9] S. U. Sambhe and S. Murab, "Literature Review on Smart Social Contact Management System for Better Memory Recall," *IJEIT*, vol. 3, no. 9, Mar. 2014. [Online]. Available: https://www.ijeit.com/Vol%203/Issue%209/IJEIT1412201403_42.pdf

[10] P. R. Saraf, "A Review on Firebase (Backend as A Service) for Mobile Application Development," *ResearchGate*, Jan. 2022. [Online]. Available: https://www.researchgate.net/publication/358244355_A_Review_on_Firebase_Backend_as_A_Service_for_Mobile_Application_Development

[11] T. K. Chawla and A. Kajala, "Transfiguring of an Android App Using Reverse Engineering," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 4, Apr. 2014. [Online]. Available: https://www.ijcsmc.com/docs/papers/April2014/V3I4201499b43.pdf

[12] freeCodeCamp, "Flutter Course for Beginners 37-hour Cross Platform App Development Tutorial," *YouTube*, 2022. [Online]. Available: https://www.youtube.com/watch?v=VPvVD8t02U8

[13] C. Fanchi, "The importance of iterative prototyping in application development," *backendless*, Feb. 7, 2023. [Online]. Available: https://backendless.com/the-importance-of-iterative-prototyping-in-application-development/#:~:text=Iterative%20prototyping%20can%20lead%20to,and%20functionality%20of%20the%20app.

[14] Net Ninja, "Flutter tutorial for beginners," *YouTube*, 2020. [Online]. Available: https://www.youtube.com/watch?v=1ukSR1GRtMU&list=PL4cUxeGkcC9jLYyp2Aoh6hcWuxFDX6PBJ