Pratibodh
A Journal for Engineering
A free and Open Access Journal
Homepage: https://pratibodh.org

UI/UX Development with Flutter

Anushika Chandel[1], Khushal Jangid[2], Anoop Sharma[3], Punita Panwar[4]
Department of Artificial Intelligence & Data Science
Jaipur Engineering College & Research Centre, Jaipur
anushikachandel.ai25@jecrc.ac.in[1], khushaljangid.ai25@jecrc.ac.in[3], anoopsharma.ai25@jecrc.ac.in[3], punitapanwar.cse@jecrc.ac.in[4]

Abstract

This research paper presents the development and evaluation of "DocTunes" a mobile application for both Android and iOS designed to ease day to day task or reading long documents. DocTunes leverages Microsoft Azure Cloud as Backend as a Service (BaaS), Google Firebase Auth for authentication, Firebase NoSQL for database, DocTunes is designed and developed using Flutter (Dart). This paper details the design, implementation, and evaluation of DocTunes, highlighting its features, user interface, and the results of user testing.

Article Status

Keywords: *Text-to-Speech, TTS, OCR, E-book reader, Flutter, Dart, Figma, Firebase*

Available online:

1. Introduction

In today's rapidly evolving, high-speed world, the existence of a substantial global community comprising 1.5 billion individuals grappling with significant language barriers when it comes to reading and comprehending their documents is a pressing concern. This monumental challenge is further compounded by the fact that over 700 million people endure the daily hardships associated with Dyslexia, rendering them incapable of independently perusing their textual materials. To address this profound need, we have taken a pioneering step by introducing a transformative and portable solution, aptly named "DocTunes."

DocTunes represents a groundbreaking initiative aimed at bridging the linguistic chasms and enhancing accessibility for those who face language impediments. This innovative project leverages cutting-edge technologies, primarily centered on the utilization of Flutter, Firebase, and the robust Azure cloud platform. By merging the powers of these digital realms, DocTunes offers a beacon of hope and inclusivity to the underserved population afflicted by language barriers and Dyslexia.

2. Methodology

Methodology describes the development process used in the development of the application. Figure 1 represents the Software Development Lifecycle used in the project. The whole process was broken down into two major steps:

2.1 Design: The designing of the application is conducted using Figma and Canva, the User Experience and User Interfaces are designed for all screens in both dark mode and light mode. The design phase essentially formulated the graphical representation of the look and feel of the application's screens such as login, signup, homepage, files, preferences etc.

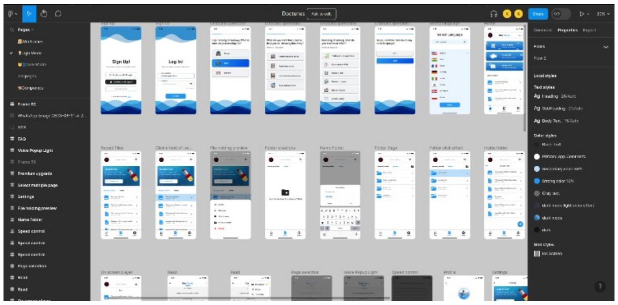

Fig. 1 UI prototypes in Figma

2.2 Implementation: The designs created in Figma are then implemented in Flutter for frontend and REST API of several Microsoft Azure Cloud & Google Cloud services are leveraged such as

Azure Translate for Translation of text, Azure TTS for text-to-speech conversion, Google Vision for text extraction from images and documents. Third party tools like RevenuCat and Mixpanel Were also used for subscription management and user tracking, respectively.

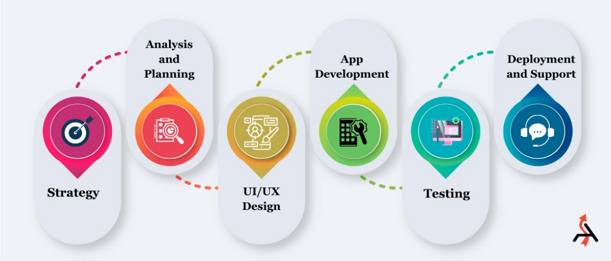

Fig. 1 App Development Lifecycle

3. Tools & Frameworks:

Figma is a platform for teams to design, prototype, develop, and collaborate in one space. It offers tools for design systems, Dev Mode, FigJam, and more**.**

Dart is an open-source programming language designed for client-side development, such as for the Web, Mobile & Desktop Applications. It is developed by Google.

Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, macOS, Windows, Linux, and the Web from a single codebase. Flutter's engine, written primarily in C++, provides low-level rendering support using Google's Skia graphics library.
Flutter has numerous advantages over any other application development technology:

- **Reduced Code Development Time:** Flutter provides a large set of Material and Cupertino widgets that perfectly mimic the behavior of each design language.
- **Similar to Native App Performance:** Application performance is crucial for good UX. Flutter app performance in most cases will be indistinguishable from the native app and even better in complex UI animation scenarios.
- **Custom, Animated UI of Any Complexity Available:** One of the biggest advantages of Flutter is the ability to customize anything you see on the screen, regardless of how complex it may be.

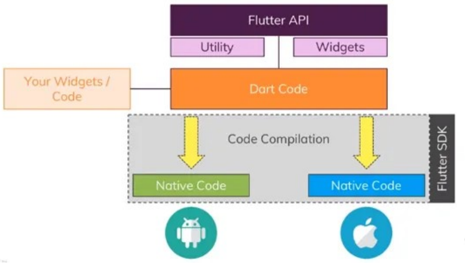

Fig. 2 Internal working of Flutter Compiler

- **Own Rendering Engine:** Flutter uses Skia for rendering itself onto a platform-provided canvas. Because of the engine, UI built in Flutter can be launched on virtually any platform.

Visual Studio Code: Visual Studio Code is code editor redefined and optimized for debugging, syntax highlighting, snippets, code refactoring etc.
Android Studio: Android Studio is the official Integrated Development Environment (IDE) for Android app development. Based on the powerful code editor and developer tools from IntelliJ IDEA, Android Studio offers even more features that enhance your productivity when building Android apps.

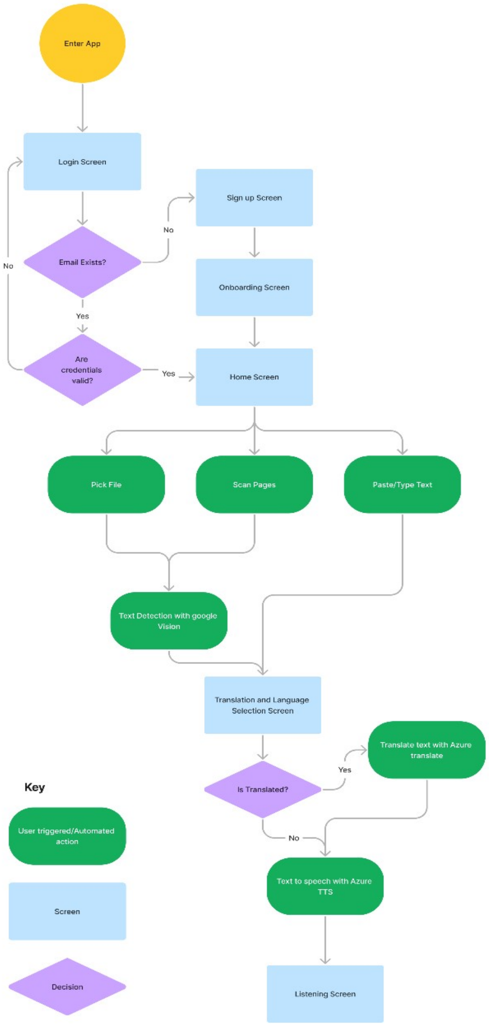

Fig. 3 Control Flow Diagram for the App

4. Architecture

The application of the application leverages Providers for state-management and runtime dependency management. User preferences are stored on the Google Firebase Firestore, while the recently translated files are stored on a local database (SQLite). After login/signup and completion of onboarding process, the user can pick a pdf or images, from the device or from the camera. These files are then stored into the cache storage of the device. These files are then converted into text using Google Vision OCR. The text is translated with Azure Translate if translation is opted otherwise, it is directly converted into speech using Azure TTS. All the cloud services (Google Vision, Azure Translate, Azure TTS) are used only for Premium users while, for free users, an offline alternative, Google's MLkit is used. The complete Control Flow from a Premium User point-of-view is shown in figure 3.

5. Future Scope

The future scope of an app like DocTunes - PDF & Text to Speech appears promising and is likely to be influenced by several factors that can shape its development and growth. Here are some aspects to consider when discussing the future scope of DocTunes:

5.1. Expanded Language and Voice Support: As the app already offers 70+ languages and 500+ voices, it can further expand its language library and voice options to cater to an even broader global audience. This would make the app more inclusive and appealing to users from diverse linguistic backgrounds.

5.2. Enhanced Accuracy and Naturalness: Continuous improvement in the quality and naturalness of speech synthesis is crucial. Advancements in AI and machine learning can help make the generated speech sound even more human-like and natural, providing a superior user experience.

5.3. Accessibility Features: As the importance of accessibility in technology continues to grow, DocTunes could focus on making its app more accessible to users with disabilities, complying with accessibility standards and guidelines.

5.4. Integration with Popular Apps and Services: Integrating with widely-used productivity and document management apps like Microsoft Office, Google Docs, and Dropbox can make it more convenient for users to access text-to-speech functionality within their existing workflows.

5.5. Education and Learning: The app can explore opportunities in the education sector. It can become a valuable tool for students, educators, and professionals in terms of content accessibility and language learning.

5.6. Security and Privacy: Ensuring the security of user data and privacy is becoming increasingly important. Implementing robust security measures and compliance with data protection regulations is essential.

6. Conclusion

The app has achieved remarkable success with over 180,000 downloads on the Play Store and users from 175+ countries. Notably, it has translated over 500 million words and converted 700,000 documents into audiobooks, showcasing its extensive functionality. Users are highly engaged, spending an average of 15 minutes using the app, with an impressive 65% retention rate. The app's steady growth is evident with a 22% month-to-month increase and a solid user base of 40,000 monthly active users. In the broader context, the market for Audiobook converters and Text-to-Speech (TTS) apps is thriving, with a remarkable 31.2% Compound Annual Growth Rate (CAGR). It is anticipated that this market will reach $15 billion by 2025, underlining the immense potential for further expansion and profitability in this industry.

References

1. Sanders, A. (2020). "Flutter in Action." Manning Publications.
2. Anderson, J., & Smith, M. (2019). "Mastering Flutter: Building High-Quality Applications." Packt Publishing.
3. Young, T., & Patil, A. (2020). "Flutter for Mobile Application Development." Springer.
4. Flutter Documentation. (https://flutter.dev/): Official documentation provided by Google for Flutter app development.
5. Gomes, C., & Silva, R. (2021). "Cross-Platform Mobile Development with Flutter." IEEE Transactions on Software Engineering.
6. Felipp, F., & Souza, D. (2018). "A Comparative Study of Cross-Platform Frameworks for Mobile App Development." In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems.
7. Bhat, V., & Varghese, B. (2020). "Performance Evaluation of Flutter and React Native for Mobile App Development." International Journal of Computer Applications.
8. Valente, F., & Figueiredo, E. (2019). "Empirical Study on the Adoption of Flutter for Cross-Platform Mobile Application

Development." In Proceedings of the International Conference on Software Engineering.

9. Liu, W., & Chen, J. (2021). "Evaluation of Flutter and React Native for Cross-Platform Mobile App Development." In Proceedings of the IEEE International Conference on Software Architecture.
10. Lawson, K., & Mitchell, J. (2018). "Building Cross-Platform Mobile Apps with Flutter." O'Reilly Media.