QUICK NOTES : A SIMPLE NOTE TAKING COMPANION

A MINI PROJECT REPORT

Submitted By

MADHULIKA G (210701139)

MAHALAKSHMI K (210701143)

MANNURU SHREEYA (210701148)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

May 2024

BONAFIDE CERTIFICATE

Certified that this project report "**QUICK NOTES: A SIMPLE NOTE TAKING COMPANION**" is the bonafide work of "**MADHULIKA G, MAHALAKSHMI K, MANNURU SHREEYA**" who carried out the project under my supervision. Certified further to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Mrs. Ananthi S

SUPERVISOR

Assistant Professor (SG) Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105.

SIGNATURE

Dr. P. Kumar

HEAD OF THE DEPARTMENT

Department Of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai – 602 105.

Submitted for Semester Mini-Project viva-voce examination held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ABSTRACT

QuickNotes is a user-friendly note-taking application developed using Android Studio, specifically designed to provide a seamless and efficient experience for users managing their notes. The application caters to both personal and professional needs by offering core functionalities such as adding, viewing, and deleting notes. Users can effortlessly create new notes with a single tap, edit existing ones to keep their information up-to-date, and view all saved notes in an organized list, providing easy access and quick navigation. The main interface of QuickNotes showcases a list of notes with titles, enabling users to quickly locate and review their entries.

Additionally, the app features a straightforward deletion mechanism, allowing users to long-press on any note to prompt a confirmation dialog for deletion. This ensures that accidental deletions are avoided, thereby providing a reliable note management experience. To store the notes, QuickNotes employs SharedPreferences, which ensures that all data is securely saved locally on the device. This approach guarantees that notes remain accessible even after the app is closed, offering a persistent storage solution without the need for external databases.

The design of QuickNotes emphasizes simplicity and ease of use, making it an ideal tool for anyone looking to effectively manage their notes on an Android device. The intuitive user interface and minimalistic design ensure that users can quickly adapt to the app and utilize its features without a steep learning curve. This project not only demonstrates the practical application of Android development skills but also showcases the ability to create functional, user-centric mobile applications. By focusing on essential features and user experience, QuickNotes stands out as a reliable and efficient note-taking solution for Android users.

ACKNOWLEDGEMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college Principal, **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. KUMAR M.E., PhD, Director computing and information science, and Head Of Department of Computer Science and Engineering** and our project coordinator **Mrs. S. Ananthi, M.TECH** for her encouragement and guiding us throughout the project towards successful completion of this project and to our parents, friends, all faculty members and supporting staff for their direct and indirect involvement in successful completion of the project for their encouragement and support.

MADHULIKA G

MAHALAKSHMI K

MANNURU SHREEYA

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

QuickNotes is a mobile application developed using Android Studio, designed to offer a seamless and efficient note-taking experience for users. In the modern world, where information overload is common, having a reliable and easy-to-use tool to capture and organize thoughts, ideas, and tasks is essential. QuickNotes aims to address this need by providing a simple yet powerful platform for users to manage their notes effectively. The app caters to a wide range of users, from students and professionals to anyone who needs a convenient way to keep track of their information.

The core functionalities of QuickNotes include adding, viewing, and deleting notes, all of which are implemented with a focus on simplicity and usability. Users can easily create new notes with a single tap, edit existing ones to update their content, and view all saved notes in an organized list format. This organized view allows for quick navigation and retrieval of information, making it easy for users to find the notes they need when they need them. The app's interface is designed to be intuitive and user-friendly, ensuring that users can get started with minimal effort and without a steep learning curve.

One of the key features of QuickNotes is its straightforward deletion mechanism. Users can long-press on any note to prompt a confirmation dialog for deletion, which helps prevent accidental deletions and ensures that users have control over their data. This feature, combined with the app's use of SharedPreferences for data storage, ensures that notes are securely saved locally on the device. This local storage solution guarantees that notes remain accessible even after the app is closed, providing users with a reliable and persistent storage option without the need for an external database.

QuickNotes represents a practical application of Android development skills and highlights the importance of creating user-centric mobile applications. The project demonstrates the ability to implement essential features in a way that prioritizes user experience and functionality. By focusing on the core needs of note-taking, QuickNotes provides a valuable tool for anyone looking to manage their notes efficiently on an Android device. This report details the development process, design considerations, and technical implementation of QuickNotes, showcasing the capabilities and features of the app, and reflecting on the lessons learned throughout the project.

1.1 PROBLEM STATEMENT

In a fast-paced world where managing information efficiently is crucial, traditional note-taking methods often fall short in providing the convenience and organizational features needed for modern productivity. There is a pressing need for a simple, intuitive, and reliable mobile application that allows users to easily create, store, organize, and delete notes on the go, ensuring seamless data storage and retrieval to enhance daily productivity. QuickNotes aims to address this gap by delivering a robust and user-friendly note-taking solution for Android devices.

1.2 SCOPE OF THE WORK

The scope of the QuickNotes project involves the development of a mobile application using Android Studio, focused on providing essential note-taking functionalities such as creating, viewing, and deleting notes. The app will feature a user-friendly interface designed for ease of use, with notes stored locally on the device using SharedPreferences to ensure data persistence. Key aspects include implementing a long-press deletion mechanism to prevent accidental deletions, optimizing performance for smooth operation, and designing a clean, intuitive user interface.

1.3 AIM AND OBJECTIVE OF THE PROJECT

The aim of the QuickNotes project is to develop a simple, intuitive, and reliable mobile application for efficient note-taking on Android devices. The primary objective is to provide users with a seamless experience for creating, viewing, and deleting notes, ensuring that their information is easily accessible and well-organized. The project seeks to design a user-friendly interface that minimizes the learning curve, implement secure and persistent local data storage using SharedPreferences, and optimize the app for performance and reliability. By addressing these objectives, QuickNotes aims to enhance users' productivity and information management in their daily lives.

1.4 RESOURCES

The QuickNotes project involved creating a mobile application using Android Studio, focusing on basic note-taking features like creating, viewing, and deleting notes. The app provides a user-friendly interface for easy use, with notes stored locally on the device using SharedPreferences for data persistence. Key features include a long-press deletion mechanism to prevent accidental deletions, performance optimization for smooth operation, and a clean, intuitive user interface design.

CHAPTER 2

LITERATURE SURVEY

Note-taking has transcended traditional methods like writing and outlining to include multimedia attachments and scanning capabilities, enriching the context and usability of notes. Various note-taking applications now offer unique features, such as extensive storage, editing, and sharing options, enhancing the user experience. The shift towards digital note-taking is evidenced by the increased adoption of electronics in educational settings, although the paper industry faces declining revenues. Interestingly, the pen and pencil industry has seen revenue growth, suggesting a complex transition where digital and traditional methods coexist. The Android Notes app, developed in Java, epitomizes modern note-taking solutions by leveraging the latest Android features to ensure a seamless user experience. It supports multimedia attachments, provides robust security with passcode and biometric options, and offers flexible file generation capabilities. These features address user needs for accessibility, security, and versatility in managing important information and daily activities. The literature underscores the ongoing integration of digital tools in education while acknowledging the persistent role of traditional writing instruments.

Apps across various domains, such as finance, education, health, life, and entertainment, enhance smartphone utility but also pose significant challenges for digital forensic analysis due to their advanced security features. These security measures, designed to protect sensitive data from external threats, necessitate sophisticated decryption techniques to render the data usable as forensic evidence. Note taking apps contain user data. However, the security mechanisms embedded within these apps, such as encryption and access controls, complicate data extraction and analysis. Studies focusing on reverse engineering and the operation processes of these security functions have been conducted to identify methods for accessing and collecting protected app data. This research aims to generalize the security features of note-taking apps and develop efficient techniques for analyzing smartphone backup data, thereby enhancing the capabilities and effectiveness of future digital forensic investigations.

The literature on multimedia note-taking systems highlights the innovative integration of digital tools to enhance meeting productivity and information capture. NoteLook exemplifies such a system, designed to facilitate multimedia note-taking through a client-server architecture that supports digital video and ink. Implemented in a

technologically advanced conference room equipped with computer-controlled video cameras and a large rear video projector, NoteLook allows users to capture and annotate meeting content interactively. The client application operates on wireless pen-based notebooks, enabling users to select video channels, incorporate background images, and add freeform ink notes. The system's smart video source management ensures high-quality image capture from various sources, while an automatic note-taking feature enhances user convenience. Web pages generated by NoteLook link ink strokes and images to corresponding video segments, facilitating easy access and browsing of notes. The development process of NoteLook involved rapid prototyping, user testing, and iterative refinements, resulting in a robust and user-friendly system. Future plans for NoteLook include further integration into meeting workflows to gather additional user experience and improve its functionality. This body of work underscores the potential of multimedia note-taking systems to transform traditional meeting documentation and enhance collaborative efficiency.

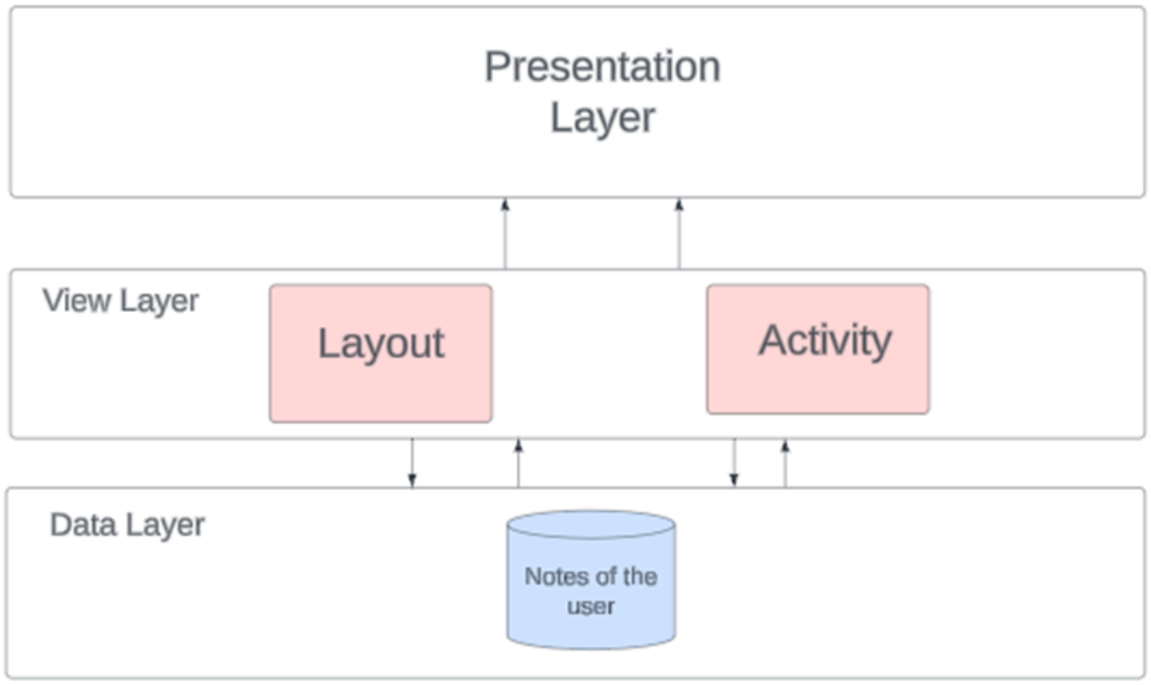
CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

In this section we would like to show how each component of the system works when organized and arranged together.

3.2 SYSTEM ARCHITECTURE DIAGRAM:



1. Presentation Layer:

· It presents the information to the user with a beautiful user interface.

· The User Interface is created using the android XML.

2. View Layer:

· It handles the activity related to each layout component.

· The activity defines the function of each component.

3. Data Layer:

· The Data Layer stores the data of the notes made by the user.

· It is deleted when the user deletes the notes.

· The data is given to other layers.

3.3 DEVELOPMENT ENVIRONMENT:

3.3.1 HARDWARE REQUIREMENTS:

The hardware requirements may serve as the basis for a contract for the system's implementation. It should therefore be a complete and consistent specification of the entire system. It is generally used by software engineers as the starting point for the system design.

Table 3.1 Hardware Requirements

COMPONENTS	SPECIFICATION
PROCESSOR	Intel Core i5
RAM	4 GB RAM
GPU	NVIDIA GeForce GTX 1650
MONITOR	15" COLOR
HARD DISK	512 GB
PROCESSOR SPEED	MINIMUM 1.1 GHz

3.3.2 SOFTWARE REQUIREMENTS:

The software requirements document is the specifications of the system. It should include both a definition and a specification of requirements. It is a set of what the system should rather be doing than focus on how it should be done. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating the cost, planning team activities, performing tasks, tracking the team, and tracking the team's progress throughout the development activity.

Android Studio version: 4.0 or higher

CHAPTER 4

PROJECT DESCRIPTION

4.1 METHODOLOGY

The methodology for the QuickNotes project was carefully structured to ensure the development of a user-friendly and efficient note-taking application for Android devices. The process began with requirements gathering, identifying the need for a simple and intuitive mobile application that enables users to create, store, organize, and delete notes efficiently. Essential functionalities, such as note creation, viewing, and deletion, were determined, and the use of SharedPreferences for local data storage was established to ensure data persistence.

Design and architecture played a critical role in the project's development. A user-friendly interface was created using Android XML, forming the visually appealing presentation layer. The view layer was defined to handle activities related to layout components and their functionalities, while the data layer was established for storing and managing notes, ensuring proper data deletion when notes were removed. Android Studio served as the primary development tool, offering a comprehensive suite for Android app development. Best practices in Android development were followed, with a strong focus on performance optimization, smooth operation, and user-centric design. The development environment was supported by robust hardware, including an Intel Core i5 processor, 4 GB RAM, and an NVIDIA GeForce GTX 1650 GPU, ensuring sufficient resources for development and testing.

Implementation focused on delivering core functionalities, emphasizing simplicity and ease of use. Key features such as note creation, viewing, and deletion were integrated, alongside a long-press deletion mechanism to prevent accidental deletions and enhance the reliability of note management. SharedPreferences was employed for data persistence, ensuring notes remained accessible even after the app was closed.

Rigorous testing and validation were conducted to identify and rectify bugs, ensuring the app's stability and performance. User testing provided valuable feedback on usability and functionality, leading to iterative improvements based on user input. This structured methodology enabled the QuickNotes project to deliver a robust, user-friendly note-taking application that enhances productivity and information management for Android users.

4.2 MODULE DESCRIPTION:

The notes app allows users to efficiently manage their notes through a simple and intuitive interface. When the user opens the app, they are greeted with a main screen (MainActivity) displaying a list of all their notes. This list is organized in a `ListView`, making it easy to browse through the notes.

From the main screen, users can add a new note by selecting the "Add note" option from the menu. This action takes them to a new screen (NoteEditorActivity), where they can enter the content of the new note using a text input field. Once the user starts typing, the note is automatically saved and updated in real-time.

The app also provides a convenient way to delete notes. By long-pressing on a note in the main list, the user is prompted with a confirmation dialog. If the user confirms the deletion, the note is removed from the list and the app's storage.

Throughout this process, the app ensures that all notes are saved persistently using shared preferences, so they are available even after the app is closed and reopened. This simple flow allows users to quickly capture, edit, and manage their notes with ease, making the app an effective tool for personal information management.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 OUTPUT

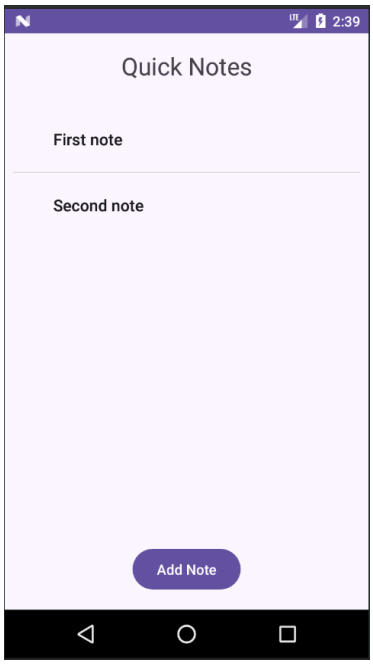


Fig 5.1 Home page with List of notes

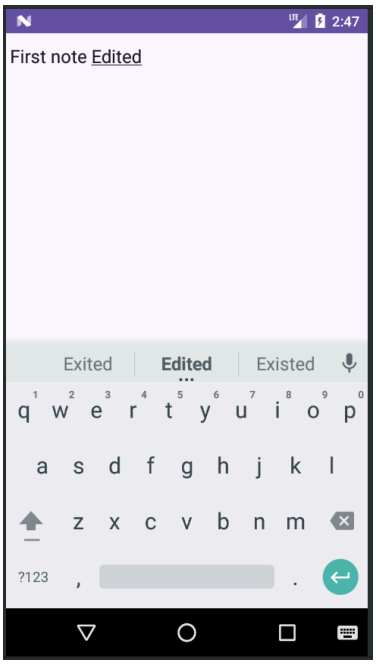


Fig 5.2 Editing a note

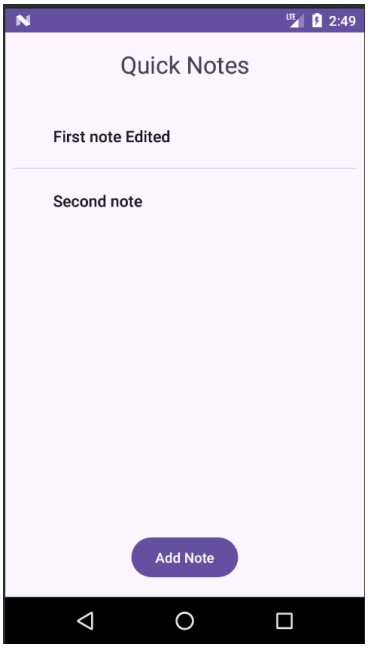


Fig 5.3 After editing a note

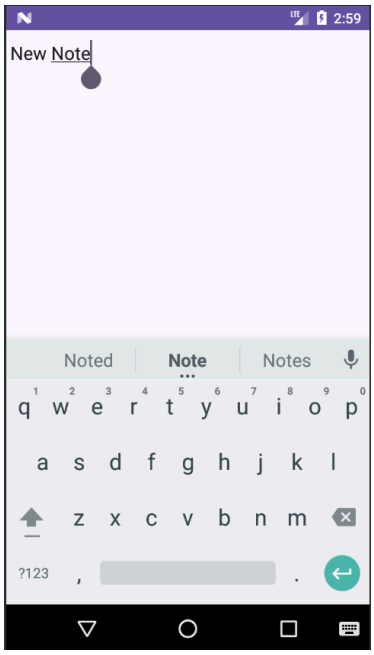


Fig 5.4 Creating new note

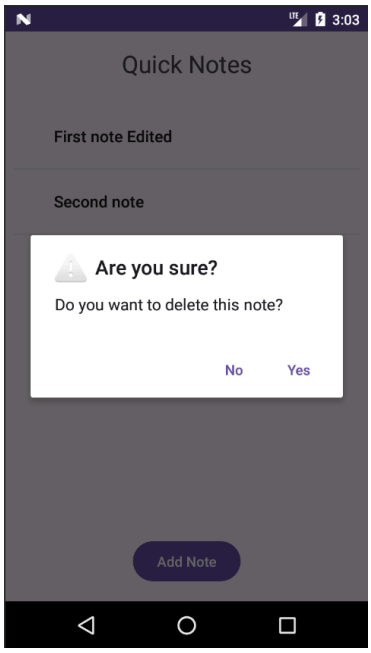


Fig 5.5 Delete functionality

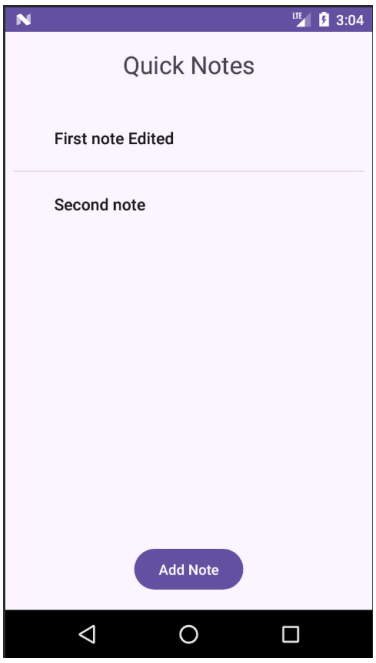


Fig 5.6 After deleting a note

5.2 RESULT

The QuickNotes project has resulted in a mobile application developed using Android Studio, focusing on essential note-taking functionalities like creating, viewing, and deleting notes. The app boasts a user-friendly interface designed for ease of use, with notes stored locally on the device using SharedPreferences for data persistence. Key features include a long-press deletion mechanism to prevent accidental deletions, performance optimization for smooth operation, and a clean, intuitive user interface. Overall, QuickNotes provides a simple yet efficient solution for users to manage their notes on Android devices.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENTS

6.1 CONCLUSION

In conclusion, the QuickNotes project successfully delivered a mobile application that meets the outlined objectives. The app enables users to efficiently create, view, and delete notes, featuring a straightforward and user-friendly interface that enhances usability. Notes are stored locally on the device using SharedPreferences, ensuring reliable data persistence. The implementation of a long-press deletion mechanism effectively prevents accidental deletions, enhancing the app's robustness. Performance optimizations have been incorporated to ensure smooth operation, and the clean, intuitive user interface provides a seamless user experience.

6.2 FUTURE ENHANCEMENTS

1. **Cloud Synchronization**: Integrating cloud synchronization capabilities will allow users to access their notes across multiple devices seamlessly. This feature will enhance the app's utility by ensuring that notes are always up-to-date, regardless of the device used.
2. **Integration with Productivity Tools**: Connecting QuickNotes with popular productivity tools like Google Drive, Dropbox, and Evernote will provide users with more options for storing and managing their notes. This integration will expand the app's functionality and convenience.
3. **Note Categorization and Tagging**: Adding features for categorizing and tagging notes will improve organization and retrieval. Users will be able to group related notes and quickly find what they need, enhancing the overall user experience.
4. **Search Functionality**: Implementing a robust search feature will allow users to easily locate specific notes by keywords or phrases. This addition will make the app more efficient, especially for users with a large number of notes.
5. **Reminders and Notifications**: Introducing a reminder and notification system will help users stay on top of their tasks and deadlines. This feature will increase the app's utility by ensuring that important notes and tasks are not forgotten.

REFERENCES

1. Dixit, Soham, et al. "Android Notes Application." *International Research Journal of Modernization in Engineering Technology and Science* 4.3 (2022): 2032-2038.

2. Park, Myungseo, Soram Kim, and Jongsung Kim. "Research on Note-Taking Apps with Security Features." *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.* 11.4 (2020): 63-76.

3. Chiu, Patrick, et al. "NoteLook: Taking notes in meetings with digital video and ink." *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*. 1999.

4. Freeman, E., Robson, E., & Bates, B. (2020). "Head First Kotlin: A Brain-Friendly Guide." O'Reilly Media.

5. Haase, C. (2019). "Android Jetpack Architecture Components: Android Jetpack Architecture Components." Apress.

6. Phillips, D. (2017). "Kotlin Programming: The Big Nerd Ranch Guide." Big Nerd Ranch Guides.

7. Martin, R. C. (2008). "Clean Code: A Handbook of Agile Software Craftsmanship." Prentice Hall.

8. Vogel, L. (2020). "Android Development Patterns: Best Practices for Professional Developers." Addison-Wesley Professional.