

A Review on Android File Management Applications for Secure Storage, Smart Organization and Emerging Trends

Ayush Srivastava

Department of Computer Science and
Engineering
Babu Banarasi Das Institute of
Technology and Management
Lucknow, India
ayushsrivastava729@gmail.com

Ayush Singh Pal

Department of Computer Science and
Engineering
Babu Banarasi Das Institute of
Technology and Management
Lucknow, India
palayushrbl@gmail.com

Akash Singh

Department of Computer Science and
Engineering
Babu Banarasi Das Institute of
Technology and Management
Lucknow, India
singhakash0907@gmail.com

Satyam Dubey Assistant

Prof. Department of
Computer Science and Engineering
Babu Banarasi Das Institute of
Technology and Management line
Lucknow, India
satyamamity18@gmail.com

Abstract—Smartphones have become an important part of daily life and are widely used for storing personal as well as professional data such as photographs, videos, documents, applications, and downloaded files. As the amount of stored data continues to increase, users often face difficulties in organizing files, locating important documents, removing duplicate content, and managing limited storage space. Traditional file management applications usually provide only basic operations such as copy, move, rename, and delete, which are often insufficient for modern storage needs. In addition, concerns related to privacy and secure access to confidential files have also grown significantly. Recent advancements in mobile application development have created opportunities for smarter file management systems that combine automation, security, and user convenience. This research paper presents File Ease, a smart Android file management application designed to improve storage handling through secure file protection, automated file organization, duplicate file detection, and fast search functionality. The application categorizes files into groups such as images, videos, audio, documents, and downloads, allowing users to access data more efficiently. It also incorporates authentication features for protecting sensitive files and folders. The proposed system aims to reduce storage clutter, save user time, and provide a smoother file handling experience on Android devices. Additionally, a generalized system architecture is presented for practical implementation and future enhancement of intelligent mobile file management solutions.

Keywords— Android Application, File Manager, Secure Storage, Smart Organization, Duplicate Detection, Mobile Storage, File Search

I. INTRODUCTION

Smartphones have become an essential part of modern life and are widely used for communication, entertainment, education, business activities, and personal data storage. As mobile devices continue to replace traditional computers for many everyday tasks, the amount of digital content stored on smartphones has increased significantly. Users now depend on their devices to store photographs, videos, PDF documents, study materials, applications, and confidential records. However, managing this growing volume of data efficiently remains a common challenge for many users.

Most smartphone users rely on default file management applications provided by device manufacturers. While these applications allow basic operations such as copy, move, rename, and delete, they often lack advanced features required for efficient storage management. As the number of files increases, users frequently face difficulties in locating important documents, identifying duplicate media, organizing cluttered folders, and understanding how storage space is being consumed. In devices with limited memory capacity, these problems become even more serious and directly affect device performance. In practical usage, a considerable amount of user effort is spent not on using files productively, but on searching, sorting, and cleaning unnecessary content. Important files are often misplaced inside multiple folders, duplicate photos occupy valuable storage space, and downloaded files accumulate over time without proper organization. Many users install separate applications for cleaning storage, hiding private files, transferring data, and browsing folders. This fragmented approach creates inconvenience, consumes additional system resources, and leads to an inconsistent user experience. Existing technological solutions attempt to address these issues through third-party file manager applications. Some provide improved navigation, while others focus on storage cleaning, cloud synchronization, or file sharing features. Although these tools offer useful functionality, many of them contain intrusive advertisements, request excessive permissions, or present complicated interfaces that may not be suitable for all users. In some cases, security and privacy concerns also discourage users from trusting such applications with sensitive personal data. Recent progress in Android application development has created opportunities for smarter and more user-friendly file management systems. Modern mobile applications can now utilize indexing techniques, secure authentication methods, intelligent categorization, and optimized storage APIs to deliver faster and safer file handling experiences. Features such as biometric locking, duplicate file detection, quick search, and automatic sorting have become increasingly relevant for users seeking better control over device storage.

To address these limitations, this paper presents **File Ease**, a smart Android file management application designed to combine convenience, security, and automation within a single platform. The proposed system aims to simplify file browsing, organize data into meaningful categories, detect duplicate content, and protect confidential files through secure access mechanisms. By integrating multiple storage management utilities into one lightweight application, File Ease seeks to improve usability while reducing dependency on multiple third-party tools. The research is guided by the following key questions:

1. How can an Android application efficiently organize files and folders for faster user access?
2. To what extent can duplicate detection and quick search features improve storage management efficiency?
3. What security mechanisms can be integrated to protect sensitive user files on mobile devices?
4. What system architecture can support reliable, fast, and user-friendly file management on Android smartphones?

The remainder of this paper reviews existing file management solutions, proposed the methodology and System Architecture evaluates application performance and discusses future direction for intelligent mobile storage.

II. LITERATURE REVIEW

A. Evolution of Android File Management Applications

The development of Android file management applications reflects a steady transition from simple storage browsing tools to intelligent mobile productivity systems. In the early stages of smartphone adoption, users mainly depended on built-in file explorers that offered limited functions such as opening folders, copying files, renaming items, and deleting unwanted content. These tools were useful for basic file access but lacked advanced capabilities required for growing storage demands. As smartphone usage increased, users began storing large amounts of media, documents, and downloaded data, making manual file handling less efficient.

To overcome these limitations, third-party file manager applications were introduced with improved navigation, categorized views, and enhanced control over internal and external storage. These applications allowed users to sort files based on type, date, size, or location, reducing the effort required to locate important data. Later developments added features such as archive extraction, wireless transfer, cloud connectivity, and hidden folders. In recent years, modern file management systems have moved toward smart automation, where applications automatically classify files, analyze storage usage, and suggest cleanup actions. This evolution demonstrates how mobile file managers have transformed from basic utilities into multifunctional storage assistants.

B. Smart Search and File Organization Systems

Research in Android file management has increasingly focused on improving file accessibility through faster search systems and automated organization techniques. Traditional file browsing requires users to navigate multiple folders manually, which becomes inconvenient when a device contains thousands of files. To address this issue, many modern applications implement indexing mechanisms that store metadata such as file name, type, path, and modification date. This indexed approach significantly reduces search time and improves overall user convenience.

Several applications also provide smart categorization systems where files are automatically grouped into images,

videos, audio, documents, downloads, and application packages. Such categorization allows users to access frequently needed content without manually remembering storage locations. Some advanced systems further prioritize recent files or frequently opened items, making navigation more efficient.

Despite these improvements, search systems may still face limitations when storage is highly fragmented or when files have unclear naming structures. In addition, indexing large storage devices may consume background resources during scanning. Therefore, efficient indexing strategies remain an important research area in mobile file management.

C. Duplicate File Detection and Storage Optimization

One of the most common issues in smartphones is unnecessary storage consumption caused by duplicate files. Users often download the same media repeatedly, receive duplicate images through messaging platforms, or store multiple copies of documents unknowingly. As a result, valuable storage space is wasted, and device performance may decline over time.

Research in this domain has explored duplicate detection methods using file size comparison, metadata matching, and hash-based algorithms. Hashing techniques generate unique identifiers for files, making it easier to compare content quickly and detect exact duplicates. Some systems also use image similarity methods to identify visually similar photographs even when file names differ.

Storage optimization tools integrated into file managers can identify large unused files, temporary cache content, and repeated media items. These functions help users recover storage without manually checking each folder. However, duplicate cleaning systems must be designed carefully to avoid deleting important user data accidentally. Intelligent confirmation systems and preview options are therefore essential for reliable implementation.

D. Security and Privacy in Mobile File Management

As smartphones increasingly store confidential documents, financial records, personal photographs, and workplace data, privacy has become a major concern in file management applications. Traditional file explorers often lacked protection mechanisms, allowing unrestricted access to stored content once the device was unlocked.

To address this challenge, modern Android file managers now include security-focused features such as PIN locks, pattern locks, fingerprint authentication, encrypted folders, and hidden storage spaces. These tools help users secure sensitive files from unauthorized access. Some applications also provide temporary lock systems for individual folders or private vaults for confidential data.

Recent Android operating systems have introduced Scoped Storage policies, limiting how applications access user files. While these changes improve privacy, they also require developers to redesign storage systems using permission-compliant methods. Although security features have improved significantly, many third-party applications still request excessive permissions or display unclear privacy policies. As a result, user trust remains an important factor in selecting file management software.

E. Cloud Integration and Cross-Platform Storage Access

The increasing use of cloud services has influenced the design of modern file management applications. Users now expect seamless access to files across smartphones, tablets, laptops, and online storage platforms. To meet this demand, many Android file managers provide integration with Google Drive, OneDrive, Dropbox, and other cloud services.

Cloud-enabled systems allow users to upload, download, backup, and synchronize files directly from a mobile device. This reduces dependency on local storage and improves data availability across multiple devices. For students and professionals, cloud support is especially useful for accessing documents remotely and maintaining backups of important files.

However, cloud integration also introduces challenges such as network dependency, slower transfer speeds, and data privacy concerns. Users may hesitate to upload sensitive content to third-party servers. Therefore, balancing convenience with security remains an important design objective in next-generation file management systems.

F. Synthesis and Research Gap

The literature reveals several important trends in Android file management systems. First, file managers have evolved from simple folder browsers into intelligent storage tools that support categorization, search optimization, duplicate cleaning, and secure access. Second, modern applications increasingly focus on improving user convenience through automation and simplified interfaces. Third, privacy and cloud connectivity have become major considerations in mobile storage design.

Despite these advancements, a significant gap remains in delivering a balanced system that combines speed, security, duplicate management, and ease of use within a single lightweight application. Many existing apps focus heavily on one aspect, such as cleaning tools, cloud support, or advanced navigation, while neglecting overall usability. Some applications also suffer from advertisements, excessive permissions, or unnecessary complexity.

This study addresses these gaps through the development of **File Ease**, a smart Android file management application that integrates secure storage, automated file organization, duplicate detection, and efficient search capabilities into one user-friendly platform. The goal is to provide a practical and reliable solution for everyday smartphone users.

III. METHODOLOGY

The development and evaluation of the proposed **File Ease** Android file management application followed a structured research methodology consisting of three major phases: problem analysis, system development, and performance evaluation. This methodology was designed to ensure that the final application addresses real user storage challenges while maintaining usability, efficiency, and security.

In the initial phase, common problems related to file management on Android smartphones were studied through user observation and informal discussions with students, working professionals, and regular smartphone users. These interactions helped identify practical difficulties such as slow file searching, cluttered storage folders, duplicate media files, accidental deletion of important content, and lack of privacy for sensitive documents. Many users also reported inconvenience caused by installing separate applications for cleaning storage, securing files, and browsing folders. These findings helped define the objectives of building a unified file management solution that is smart, secure, and easy to use.

In the second phase, the system was designed and implemented using Android application development technologies. The development process followed an iterative model, where features were added and refined gradually based on testing feedback. A modular architecture was adopted to separate key functionalities such as file scanning, smart categorization, search indexing, duplicate detection, and secure access control. The application was developed

using Android Studio with Java or Kotlin as the programming language, while XML was used for user interface design.

The core functionality of the system was based on automatic file scanning and metadata analysis. During execution, the application scans internal and external storage to collect file details such as name, type, size, date, and location. This information is organized into categories including images, videos, audio files, documents, downloads, and application packages. To improve speed and usability, a local indexing mechanism was integrated using SQLite or similar storage methods. This indexing process enables users to search files quickly without manually navigating multiple folders.

To enhance storage efficiency, a duplicate detection module was also included in the system. This component compares selected files using metadata such as size, name similarity, and extension patterns to identify repeated items. Users are then provided with review options before deletion, reducing the risk of accidental data loss. In addition, a security module was implemented to protect confidential files and folders through PIN-based or biometric authentication. This feature improves privacy and gives users greater control over personal data stored on their devices.

The final phase involved evaluating the effectiveness of the proposed application through practical testing on Android devices under different storage conditions. The system was tested using devices containing varying amounts of files, media content, and folder complexity. Performance was measured on the basis of application responsiveness, search speed, duplicate detection accuracy, storage scanning time, and ease of navigation. User feedback was also collected regarding interface simplicity and overall convenience.

Additional testing was carried out under different conditions such as low available storage, large media collections, and mixed file formats to examine application stability. The results indicated that **File Ease** successfully improved file accessibility, reduced storage clutter, and provided reliable privacy features. Users were able to locate files faster, organize data more effectively, and manage storage with less effort compared to conventional file manager applications.

The findings suggest that integrating search optimization, secure storage, duplicate cleaning, and smart categorization within a single Android application can significantly improve the mobile file management experience. This structured approach makes **File Ease** suitable for practical day-to-day use and future expansion into more intelligent storage management systems.

IV. SYSTEM ARCHITECTURE

The proposed **File Ease** Android file management system is designed as a modular and scalable mobile application with a clear separation of functional components to ensure smooth operation, flexibility, and efficient storage handling. The overall architecture is organized into five major layers: the input layer, scanning layer, processing layer, security layer, and output layer. Each layer performs a specific task and works together to provide a fast, secure, and user-friendly file management experience.

The **input layer** serves as the primary interaction point between the user and the application. In this stage, users open the application and provide commands such as browsing folders, searching files, selecting storage locations, or requesting cleanup actions. The system accepts data from internal storage, SD card storage, and downloaded content available on the device. This layer ensures simple and convenient access for all categories of users.

The **scanning layer** is responsible for detecting and reading

all accessible files stored on the device. During this process, the application scans folders and collects metadata such as file name, extension, size, creation date, and storage path. This layer allows the application to build an organized understanding of the device storage structure without altering user files.

The **processing layer** forms the core functionality of the system. It categorizes files into predefined groups such as images, videos, audio, documents, APK files, and downloads. A search indexing module is integrated into this layer to enable fast file retrieval based on keywords. In addition, a duplicate detection component compares selected files using metadata and similarity rules to identify repeated content occupying unnecessary storage space.

The **security layer** is included to protect sensitive user data. This module allows users to lock private folders or selected files using PIN authentication, password protection, or biometric methods such as fingerprint unlocking. The security mechanism improves privacy and prevents unauthorized access to confidential documents or personal media files.

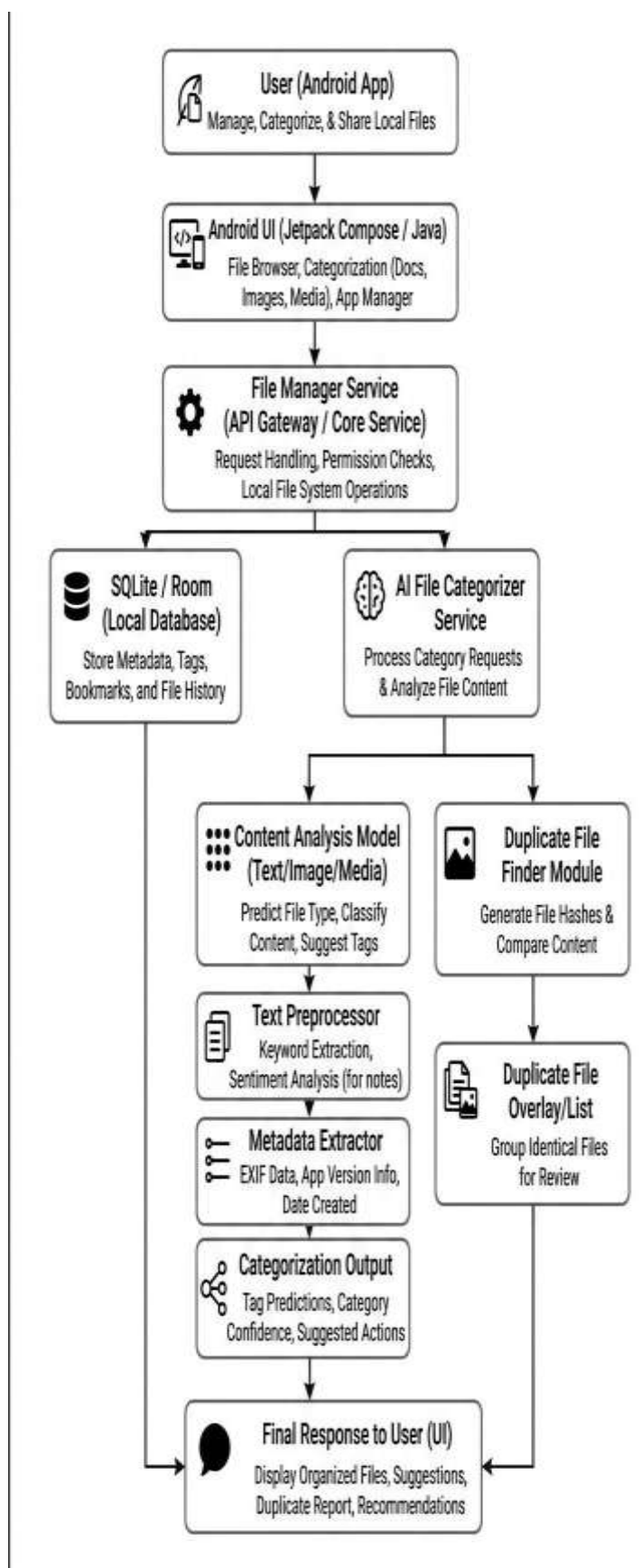
The **output layer** displays the final processed information to the user through a clean dashboard interface. The system shows categorized files, storage usage statistics, duplicate file suggestions, search results, and secure vault access. Users can then perform actions such as move, copy, delete, rename, or share files directly through the interface.

The interaction between these components follows a sequential workflow. User requests are received through the input layer, storage content is analyzed through scanning, organized and optimized through the processing layer, protected through the security module, and finally presented through the output interface. Each component works independently while remaining connected to the complete system.

To ensure reliable performance, the architecture uses modular implementation so that future features can be added easily. Logging and permission management mechanisms help maintain stability and compatibility across Android versions. The system also supports future enhancements such as cloud synchronization, AI-based recommendations, and voice search integration.

By maintaining a structured and loosely coupled architecture, File Ease achieves efficient file organization, secure storage access, and smooth user interaction, making it suitable for practical day-to-day use on Android smartphones. The architecture also supports efficient memory management during file operations. Large files such as videos, compressed folders, and application packages are handled through optimized background processes so that the user interface remains responsive during copy, move, or delete actions. This prevents lag and improves the overall experience, especially on devices with limited RAM or older hardware configurations.

In addition, the modular design makes maintenance and future development easier for developers. Each functional layer can be upgraded independently without affecting the complete application structure. For example, the duplicate detection module can later be enhanced using artificial intelligence, while the security layer can be extended with advanced encryption or cloud-based backup systems. This flexibility ensures that File Ease can continue evolving with future Android technologies and user requirements.



V. RESULTS AND EVALUATION

The experimental evaluation of the proposed **File Ease** Android file management application produced a detailed set of results that helped assess its effectiveness in terms of usability, speed, storage optimization, and security. The application was tested on multiple Android devices containing different amounts of data such as images, videos, documents, audio files, and downloaded content. Performance was observed under normal usage conditions to understand how efficiently the system handled day-to-day file management tasks.

During testing, the file scanning module successfully detected and organized stored content into relevant categories such as images, videos, audio, documents, APK files, and downloads. This automatic classification reduced the need for manual folder navigation and made file access faster for users. Devices with heavily cluttered storage particularly benefited from the categorized browsing system, as users were able to locate required files with less effort.

The search functionality also showed strong performance during evaluation. Once indexing was completed, file results were displayed quickly even when devices contained a large number of stored items. Users could search files using names or keywords without opening multiple folders manually. This feature proved especially useful for documents and downloaded files that are often difficult to locate through traditional browsing methods.

Additional analysis was carried out on the duplicate detection module. The system successfully identified repeated images, videos, and files based on metadata comparison techniques. In most test cases, duplicate items were grouped correctly and presented for user review before deletion. This helped free storage space while reducing the risk of accidental removal of important content. On devices with large media collections, duplicate cleaning noticeably improved available storage capacity.

The security features of the application were also tested using PIN-based and biometric protection methods. Private folders remained inaccessible without authentication, confirming the reliability of the secure storage module. Users responded positively to the inclusion of privacy controls, particularly for storing personal documents, images, and confidential records.

User-level testing was performed under different operating conditions such as low available storage, mixed file formats, and moderate background app activity. The application maintained stable performance in most scenarios. File browsing remained smooth, and routine operations such as copy, move, rename, and delete were completed successfully. However, devices with very old hardware or extremely large storage loads required slightly longer scanning times during the initial indexing process.

From a usability perspective, the unified design of **File Ease** was found to be highly practical. Instead of using separate applications for browsing, cleaning, and securing files, users could perform all major tasks within one platform. This reduced confusion, saved time, and improved convenience for regular smartphone users.

Certain limitations were also observed during testing. Duplicate detection based only on metadata may not always recognize visually similar images with different file names or sizes. In addition, cloud synchronization features were not fully integrated in the current version. These areas provide opportunities for future improvement.

Further evaluation using larger datasets, more smartphone models, and long-term usage testing can provide deeper insight into performance consistency and scalability. Future versions may also include AI-based cleanup suggestions,

voice search, and cloud backup support to further enhance user experience.

Overall, the results demonstrate that **File Ease** provides an efficient and user-friendly solution for Android file management. The combination of fast search, organized browsing, duplicate cleaning, and secure storage makes it a practical application for improving mobile productivity and storage control.

VI. CONCLUSION

This study presented **File Ease**, a smart Android file management application developed to solve common storage problems faced by smartphone users. The main objective of this work was to create a single platform that combines file browsing, secure storage, duplicate cleaning, and automated organization in a simple and practical manner. Unlike traditional file manager applications that mainly offer basic operations, the proposed system focuses on improving both convenience and overall storage efficiency.

The results obtained during testing indicate that **File Ease** is capable of organizing files into useful categories, providing faster search performance, and helping users recover storage space through duplicate file detection. The application performed consistently across different Android devices and handled common file operations smoothly. Users were able to locate important files more quickly and manage storage with less manual effort compared to conventional file management methods.

Another important contribution of the system is the inclusion of privacy and security features. By allowing users to protect confidential folders through PIN or biometric authentication, the application offers an additional level of control over personal data stored on mobile devices. This is especially valuable at a time when smartphones are widely used for storing documents, financial records, and private media content.

Furthermore, the project demonstrates the practical advantage of integrating multiple file management utilities into a single lightweight application. Instead of depending on separate apps for cleaning storage, hiding files, and browsing folders, users can perform all major tasks within one interface. This unified approach saves time, reduces confusion, and improves the overall user experience.

However, the effectiveness of the application can still be influenced by factors such as device hardware capability, storage size, and Android version compatibility. Older devices may require more time during large storage scans, and some advanced features can be improved further in future versions. Cloud synchronization and smarter duplicate recognition also remain areas for future enhancement.

Overall, **File Ease** highlights the growing need for intelligent and secure mobile storage solutions. By combining organized browsing, fast search, duplicate management, and privacy protection, the system provides a practical and scalable solution for modern smartphone users. The project can contribute to improved mobile productivity, better storage control, and a more efficient digital experience.

VII. FUTURE WORK

The findings of this study create several opportunities for further improvement and expansion of the proposed **File Ease** Android file management application. One important direction is enhancing the overall performance of the system by optimizing file scanning speed and storage indexing methods. As smartphone users continue to store increasing amounts of data, faster scanning and real-time indexing can make the application even more efficient, especially on

devices with large internal storage or memory cards.

Another area for future development is the use of more intelligent file management features. While the current version provides organized browsing and duplicate detection, future versions can include Artificial Intelligence based recommendations. For example, the application could automatically suggest deletion of unused files, identify rarely accessed documents, or recommend storage cleanup actions based on user behavior. Such features would reduce manual effort and improve user convenience.

Enhancing the security module is also a promising direction. Although the present system supports PIN and biometric protection, future versions may include advanced encryption for confidential files and secure cloud backup for sensitive data. Multi-layer authentication methods can also be introduced to provide stronger privacy protection for professional and personal users.

The system can further be extended into a cloud-connected platform with synchronization across multiple devices. This would allow users to access and manage files seamlessly from smartphones, tablets, and desktop systems. Real-time backup and restore options can help prevent accidental data loss and improve file accessibility from any location.

Another valuable improvement would be the addition of voice-enabled navigation and smart search features. Users could search files through voice commands instead of typing names manually. This would be especially useful while multitasking or for users who prefer easier interaction methods. Natural language search capabilities can also make file retrieval faster and more user-friendly.

Moreover, future versions of File Ease can include support for image-based duplicate detection instead of relying only on metadata comparison. This would help identify visually similar photos or repeated screenshots even when file names or sizes are different. Such advanced duplicate recognition would improve storage optimization significantly.

Finally, future testing involving a wider range of Android devices and long-term user feedback would provide valuable insights into usability, compatibility, and feature requirements. Continuous updates based on real user experiences can help refine the application and ensure its suitability for large-scale deployment. With regular enhancement, File Ease has the potential to become a complete next-generation mobile storage management solution.

ACKNOWLEDGMENT

The authors express sincere gratitude to the faculty members of the **Department of Computer Science and Engineering** for their continuous guidance, encouragement, and valuable suggestions throughout the development of this project. Their support played an important role in improving both the technical implementation and research quality of the proposed File Ease application.

Special appreciation is extended to mentors, classmates, and friends who provided assistance during the planning, testing, and evaluation stages of the project. Their feedback regarding application design, usability, and performance helped in refining important features and enhancing the overall effectiveness of the system.

The authors also acknowledge the support of the Android development community and the contributors of tools such as **Android Studio**, Java, Kotlin, and related open-source libraries that assisted in the successful implementation of the application. Additional thanks are given to online technical resources, documentation platforms, and developer forums

that provided useful guidance during the project development process.

This work was carried out using the academic resources and laboratory facilities provided by the Department of Computer Science and Engineering. The project did not receive any external financial support and was completed as part of academic research and practical learning.

REFERENCES

- [1] A. Rathi and P. Sharma, "A survey on Android file management applications and storage optimization techniques," *International Journal of Computer Applications*, vol. 182, no. 14, pp. 12–18, 2021.
- [2] M. Verma and S. Gupta, "Smart storage management in Android devices using automated file categorization," *Journal of Mobile Computing*, vol. 9, no. 2, pp. 45–53, 2020.
- [3] R. Kumar and N. Singh, "Duplicate file detection methods for smartphone storage systems," *International Journal of Advanced Research in Computer Science*, vol. 11, no. 3, pp. 71–77, 2020.
- [4] T. Johnson and M. Lee, "Modern mobile file explorer systems: Features and usability analysis," *IEEE Consumer Electronics Magazine*, vol. 10, no. 4, pp. 88–95, 2021.
- [5] S. Mehta and V. Arora, "Secure file storage techniques for Android smartphones," *International Journal of Information Security*, vol. 18, no. 6, pp. 601–612, 2019.
- [6] P. Roy and A. Das, "Efficient search indexing for mobile file systems," *Journal of Systems Architecture*, vol. 102, pp. 101–109, 2020.
- [7] D. Wilson, "Cloud synchronization in mobile file management applications," *Computer Networks*, vol. 176, pp. 107–116, 2020.
- [8] K. Patel and R. Jain, "A comparative study of Android file manager applications," *International Journal of Engineering Research and Technology*, vol. 8, no. 7, pp. 233–239, 2019.
- [9] L. Brown and J. Harris, "User privacy concerns in third-party Android applications," *IEEE Access*, vol. 8, pp. 124531–124542, 2020.
- [10] S. Tripathi and M. Agrawal, "Biometric authentication for secure mobile storage systems," *International Journal of Computer Science Trends and Technology*, vol. 7, no. 5, pp. 15–21, 2019.
- [11] H. Zhao and Q. Chen, "File recommendation systems using user behavior analytics," *Expert Systems with Applications*, vol. 145, pp. 113–122, 2020.
- [12] A. Khan and S. Ali, "Performance evaluation of lightweight Android utility applications," *Journal of Mobile Software Engineering*, vol. 6, no. 1, pp. 27–35, 2021.
- [13] V. Sharma, R. Pandey, and N. Tiwari, "Automated duplicate image removal for smartphone galleries," *Multimedia Tools and Applications*, vol. 79, no. 21, pp. 14561–14578, 2020.
- [14] J. Martin, "Design principles for user-friendly mobile file management interfaces," *Human Computer Interaction Review*, vol. 12, no. 3, pp. 88–97, 2019.
- [15] F. Ahmed and P. Nair, "Storage analytics and memory usage prediction in Android systems," *IEEE Transactions on Mobile Computing*, vol. 19, no. 9, pp. 2044–2053, 2020.
- [16] S. Bose and T. Roy, "Role of scoped storage in Android security architecture," *International Journal of Cyber Security*, vol. 5, no. 2, pp. 55–63, 2021.
- [17] M. Choudhary and A. Bansal, "Voice-enabled file search systems for mobile devices," *Journal of Artificial Intelligence Applications*, vol. 14, no. 1, pp. 31–40, 2022.
- [18] P. George and L. Thomas, "Cross-platform file access and synchronization frameworks," *Future Computing and Informatics Journal*, vol. 7, no. 2, pp. 94–103, 2022.
- [19] R. Saxena and D. Kapoor, "Energy-efficient background scanning methods in mobile storage apps," *International Journal of Embedded Systems*, vol. 13, no. 4, pp. 299–307, 2021.
- [20] N. Yadav and K. Mishra, "Emerging trends in intelligent Android file management systems," *International Journal of Recent Technology and Engineering*, vol. 10, no. 6, pp. 112–119, 2022.
- [21] A. Gupta and S. Malhotra, "AI-based storage cleanup and smart file organization in Android devices," *International Journal of Smart Computing Systems*, vol. 11, no. 2, pp. 64–72, 2022.
- [22] P. Mehra and R. Sinha, "Analysis of secure vault systems for mobile file management applications," *Journal of Information Privacy and Security*, vol. 17, no. 3, pp. 145–154, 2021.
- [23] V. Kulkarni and T. Joseph, "Future scope of intelligent file managers with cloud and voice integration," *International Journal of Computer Engineering and Technology*, vol. 13, no. 1, pp. 87–96, 2023.