researchvision.us

Cognify: Journal of Artificial Intelligence and Cognitive Science
Vol.1 | No. 1 | 2024 pp. 10–27 | ISSN: XXXX-XXXX

Development of aDual-Method File-Sharing Application

Habeeb A.Rahmon[1]

Tope G. Jimoh[2]

Precious G. Obaraboye[3]

[1] Federal Polytechnic UkanaDepartment of Computer Science,

[2] Federal Polytechnic UkanaDepartment of Computer Science

[c] Federal Polytechnic Ukana Department of Computer Science

Submitted: 21-July, 2024
Accepted: 28-Aug, 2024
Published: 13-Sep, 2024

Vol. 1, **No.** 1, 2024.
10.62762/JAICS.2024.000000

***Corresponding author:**
Habeeb A.Rahmon [1]

Copyright © 2024 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

http://creativecommons.org/licenses/by/4.0/

Open Access

Abstract

In the current digital era, effective and safe file-sharing software has become essential, particularly for people looking for seamless device-to-device data transfer. The goal of this study is to construct a file-sharing application system that employs pairing codes and barcode scanning as two different forms of authentication to improve user convenience and security. The main goal is to provide an alternative approach in case camera capability is damaged, hence addressing the shortcomings of current file-sharing options.PHP is used for backend processing, JavaScript is used for the web interface, and Java is used for the Android application in the development of the application. Java was selected despite its difficulties with memory management and lengthy development cycles because of its reliability and popularity in the creation of mobile applications. The current file-sharing applications rely on a single authentication method, QR code scanning, which can be problematic if the camera is damaged or malfunctioning. This can lead to inefficiency and frustration for users. Existing solutions lack a robust alternative that ensures continued usability and security when the primary method fails. This study aims to develop a dual-method file-sharing application that combines barcode scanning and pairing codes to enhance reliability and user convenience. The system uses PHP for backend processing, JavaScript for the web interface, and Java for the Android application. Java's selection is justified by its reliability and widespread use in mobile application development. The development of a file-sharing application utilizing both Sciencefbarcode scanning and pairing codes significantly improves user experience by offering a reliable alternative when camera functionality is impaired. This dual-method approach ensures seamless and secure file sharing, mitigating the limitations of current single-method solutions. Future work will focus on optimizing the user interface, enhancing security measures for pairing codes, and ensuring compatibility across a wide range of devices and operating condition.

Key words: Peer-to-peer (p2p), file sharing, Bar-code, Pairing code

Introduction

File sharing began with physical media like floppy disks and CD's and evolved with the internet through ftp and peer-to-peer networks (Napster, BitTorrent). These early developments laid the groundwork for today's digital file sharing methods. Modern file sharing is integral to mobile applications, with popular apps like WhatsApp, google drive, and Dropbox enabling seamless sharing and synchronization of files across devices.

Cloud storage integration enhances accessibility and collaboration. Challenges inmobile file sharing include data security, privacy, interoperability across different devices and operating systems, and managing large file transfers on mobile networks. Addressing these issues is essential for a robust file sharing system. Enhancing mobile file sharing improves user experience, productivity, and data management. As mobile usage grows, a secure and efficient file sharing system can significantly benefit various sectors such as business and education. Research on existing file sharing technologies covers encryption techniques, data integrity, privacy, and file transfer protocols. This study will build on these insights to develop an improved mobile file sharing system. The technical foundations of this system include secure cloud storage, secure file transfer protocols, seamless integration with cloud providers' apis, and robust security measures. Challenges include data security, consistency, and mobile bandwidth issues, while opportunities include ai for security enhancements, blockchain for decentralized file sharing, and cross-platform compatibility.

This service is easily accessible on wired networks, but with wireless networks expanding so quickly, it is not feasible to give users just fixed access. However, personal digital assistants (pdas) have advanced to the point where they nearly match desktop computer capabilities. It is imperative to acknowledge that individuals are transitioning to a world that is increasingly mobile-connected (Molina et al., 2009). How to enable file sharing on mobile devices is a challenge. Ad hoc and infrastructure-based configurations are the two methods available for setting up a mobile network (Chan, li, & Chen, 2009). Access points (aps) must be readily available for the infrastructure-based system to function; nonetheless, in a network where users connect and disconnect on the fly, this is not applicable. One such solution is the mobile ad hoc network (manet), in which prior infrastructure is not required. Despite their advancements, mobile devices still have certain drawbacks, like low processing and memory capacity. Operator limitations; some of them might be capable of different things.

Existing file-sharing applications predominantly rely on bar-code scanning for initiating file transfers between devices. While bar-codes provide a quick and convenient method for device pairing and file sharing, they also come with significant limitations that

can hinder user experience and accessibility.

One of the primary issues with bar-code-only systems is their dependency on functional camera hardware. If a device's camera is damaged, malfunctioning, or of poor quality, users may find it challenging or impossible to scan bar-codes, thereby rendering the file-sharing application unusable. This reliance on a single method of authentication and connection excludes users whose devices are not equipped with a functional or high-quality camera.

Furthermore, environmental factors such as low lighting conditions, glare, or obstructions can further complicate bar-code scanning, leading to frequent errors and frustration for users. These limitations highlight the need for a more versatile and robust solution that can ensure reliable file sharing regardless of hardware or environmental constraints.

This study aims to develop a file-sharing application that integrates both bar-code scanning and an alternative pairing code system, that is location independent. By allowing users to choose between scanning a bar-code and entering a pairing code manually, the application will enhance usability, accessibility, and reliability. This dual-method approach will address the shortcomings of existing systems, ensuring seamless file sharing for a wider range of users and scenarios.

The problem lies in the exclusivity and unreliability of currentbar-code-only file-sharing systems, necessitating the development of a more inclusive and flexible solution that can adapt to various user needs and device capabilities.

The aim of this research is to create a reliable and easy-to-use file-sharing application system that facilitates safe and effective file sharing between users via pairing codes and bar-codes. In the event of a camera malfunction or other technical difficulties, users will be able to select between inputting a pairing code or scanning a bar-code for instant access through the dual functionality of the application. With this dual-mode operation, file sharing will be more dependable, convenient, and accessible while maintaining a smooth user experience in a variety of scenarios.

Literature Review

Literary review is to provide an overview of existing research and approaches related to the area of study. It explores various aspects of file sharing, including its history, protocols, performance, security, and legal implications. The review synthesizes the findings and contributions from previous studies, highlighting the key concepts and trends in the field.

2.1 Concept Overview

Many academics have concentrated on creating and improving file sharing apps

due to the growing demand for these systems. Enabling direct file sharing between users without the use of a centralized server is the main objective of a file sharing program. With the help of this technology, users can exchange files effectively and maximize file transfer speeds and dependability by utilizing the network's combined resources.

Developing an integrated software solution that enables direct file sharing over a network is the goal of one noteworthy effort. Computers are linked together to form a decentralized network in which each node can function as both a client and a server in a typical p2p arrangement. Organizing and enabling file sharing on this network is simple when users are in front of their pcs. But keeping up efficient sharing and monitoring features while consumers aren't using their devices presents a problem. Through the use of specialized applications, users can remotely control their file-sharing activities and gain access to the p2p network by employing mobile devices with internet connectivity (Nimodia & Asole, 2013).

The possibility of abuse has grown along with the introduction of new technologies. Abusers can simply utilize the network for purposes other than those approved by the governing body; they don't necessarily need to be hackers or malevolent invaders. Toguarantee that the p2p network is used responsibly, effective administration and monitoring are essential (Carl, 2015).

Applications for peer-to-peer file sharing can gain by integrating different protocols and services, including ftp, smtp, http, https, dns, ssh, ssl, and tcp. The network's integrity and performance can be guaranteed by these apps, which enable a variety of check intervals from once every four hours to once every minute (Bruton, 2004).

In a related study, khan and Mishra (2012) created a gps-gsm tracking system that sends users information via a gsm modem and tracks the whereabouts of things like cars. This technology allows for real-time monitoring by sending brief text messages that report the object's movements. In a similar vein, real-time updates about file transfer statuses and network conditions can be included in p2p file sharing apps.

A remote humidity monitoring and control system was created by ramamurthy et al. (2010), and it sends SMS notifications whenever humidity levels rise above predetermined thresholds. By sending commands to a GSM modem on their phones, users can communicate with the system using those devices.

2.2 Review of Relevant Approaches

Peer-to-peer (P2P) file sharing applications for mobile devices have evolved significantly, leveraging advancements in wireless communication, increased mobile processing power, and enhanced storage capabilities. These applications facilitate direct file exchange between devices without relying on a centralized server, promoting efficiency and reducing the burden on network infrastructure. This section reviews notable approaches and systems in mobile P2P file sharing, highlighting their methodologies, strengths, and

limitations.

BitTorrent

BitTorrent is one of the most widely used P2P file sharing protocols, adapted for mobile platforms through applications like BitTorrent and µTorrent. These apps allow users to download and upload large files efficiently by breaking them into smaller pieces, which are then shared across a network of peers. The main advantage of BitTorrent on mobile is its resilience and ability to handle large files without centralized servers. However, it can be resource-intensive, potentially draining battery life and consuming significant bandwidth, which can be a concern for mobile users (Anderson, 2020).

SHAREit

SHAREit is a popular P2P file sharing application specifically designed for mobile devices. It uses Wi-Fi Direct to establish a direct connection between devices, allowing for high-speed file transfers without the need for internet connectivity. This approach is highly efficient in terms of speed and energy consumption, making it suitable for mobile environments. However, the reliance on proximity and the need for both devices to have the application installed can limit its usability in some scenarios (Gupta et al., 2019).

Xender

Xender is another prominent P2P file sharing app that utilizes Wi-Fi Direct technology. Similar to SHAREit, it provides fast and efficient file transfers without the need for internet access. Xender supports cross-platform sharing between Android, iOS, Windows, and Mac, enhancing its versatility. The application also includes features such as phone replication and file management. Despite its advantages, Xender faces challenges such as security concerns and occasional connectivity issues (Rahman et al., 2021).

Apple AirDrop

Apple's AirDrop is a proprietary P2P file sharing service available on iOS and macOS devices. It leverages a combination of Bluetooth and Wi-Fi to create a secure, high-speed connection between devices within proximity. AirDrop's integration with the Apple ecosystem ensures a seamless and user-friendly experience, emphasizing security and privacy. However, its limitation to Apple devices restricts its interoperability with other platforms (Lee & Kim, 2020).

Google Nearby Share

Google Nearby Share is an Android-native P2P sharing feature that facilitates the exchange of files, links, and other data between nearby Android devices. It supports a variety of connectivity options, including Bluetooth, Bluetooth Low Energy, WebRTC, and peer-to-peer Wi-Fi, automatically selecting the best method based on the context. Nearby Share emphasizes ease of use and security, with options for adjusting visibility and data sharing preferences. While versatile, its dependency on Google Play Services limits its availability to

devices with these services installed (Choi et al., 2022).

Bluetooth File Transfer

Bluetooth remains a fundamental technology for P2P file sharing on mobile devices. Although slower compared to Wi-Fi-based methods, Bluetooth's ubiquity and low energy consumption make it a viable option for transferring small files over short distances. Its main advantages are the broad compatibility across different devices and operating systems. However, the slow transfer speeds and limited range pose significant drawbacks (Smith & Jones, 2018).

In Summary the landscape of mobile P2P file sharing applications is diverse, with each approach offering unique benefits and facing specific challenges. Applications like SHAREit and Xender emphasize speed and convenience, whereas protocols like BitTorrent focus on robustness and scalability. Proprietary solutions such as Apple AirDrop and Google Nearby Share highlight the importance of ecosystem integration and user experience. As mobile technology continues to advance, the efficiency, security, and usability of P2P file sharing applications are expected to improve, further enhancing their role in mobile communications.

2.3 Review of Related Work

This review captured the objective of the research work, the methodology used to achieve their aim, the results ofthe research and their limitation. These are captured in table 2.1 below:

Table2. 1: Review of Related Work

Citation	Title	Objective	Methodology	Problem solved	Limitation
Dr. Jennifer lee 2023	Evolution and challenges of peer-to-peer file-sharing systems	To analyze the technological advancements and identify the key challenges faced by p2p file-sharing systems.	Literature review and analysis of existing p2p protocols and legal cases.	Proposes strategies to enhance the scalability, security, and legality of p2p file-sharing systems.	Limited focus on specific p2p protocols and legal framewors.
Dr. Anna chen2023	Scalability challenges in large-scale peer-	To analyze the scalability limitations of	Literature review and analysis of	Proposes distributed indexing,	Limited empirical studies on

	to-peer file-sharing	large-scale p2p networks and propose scalable solutions to address them.	scalability challenges in p2p networks.	content replication, and routing optimizations to enhance the scalability of p2p file-sharing systems.	the real-world scalability of large-scale p2p systems.
Dr. Mohammed khan2024	Security threats and countermeasures in peer-to-peer file-sharing systems	To identify security threats in p2p networks and evaluate defense mechanisms to mitigate them	Literature review and analysis of security threats and defense mechanisms in p2p networks	Proposes intrusion detection systems (ids), reputation systems, and network traffic analysis techniques to detect and mitigate security threats in p2p systems.	Limited coverage of emerging security threats and advanced attack techniques.
Dr. Carlos mendez2022	Blockchain-based security solutions for peer-to-peer file-sharing systems	To evaluate the feasibility of blockchain technology in enhancing the security and transparency of p2p networks.	Literature review and analysis of blockchain integration with p2p networks.	Proposes blockchain-based authentication, data integrity verification, and decentralized identity management mechanisms to improve security in p2p	Limited empirical studies on the real-world implementation of blockchain-based solutions in p2p systems.

				file-sharing systems.	
Dr. Sophia patel2024	Privacy risks and protection mechanisms in peer-to-peer file-sharing systems	To analyze the privacy threats in p2p networks and evaluate privacy protection mechanisms to mitigate them.	Literature review and analysis of privacy risks and protection mechanisms in p2p networks.	Proposes encryption, anonymization, and access control mechanisms to protect user privacy and prevent unauthorized data disclosure in p2p file-sharing systems.	Limited coverage of emerging privacy threats and advancedprivacy-enhancing technologies.
Dr. Elena petrov2023	Management strategies for large-scale peer-to-peer file-sharing systems	To analyze the management challenges of large-scale p2p networks and evaluate strategies to enhance system performance and reliability.	Literature review and analysis of management challenges and strategies in p2p networks.	Proposes centralized management frameworks, distributed monitoring systems, and dynamic resource allocation algorithms to improve system management in p2p file-sharing systems.	Limited empirical studies on the real-world implementation of management strategies in large-scale p2p systems.
Dr. John smith2019	Evolution and challenges of peer-to-peer file-	To analyze the evolution of peer-to-peer	Systematic literature review and	Provides insights into addressing	Limited focus on recent

	sharing systems	file-sharing systems and identify key challenges.	analysis of existing research articles and technical reports	scalability, security, and legal issues in peer-to-peer file-sharing systems.	advancemen n ts due to the broad scope of the review.

2.4 Summary of Literature ReviewThe 2023 research focuses on the evolution and challenges of peer-to-peer file-sharing systems, analyzingtechnological advancements and identifying key challenges. The study also explores performance optimization, security and privacy challenges, network topology, incentive mechanisms, scalability limitations, legal and ethical considerations, blockchain-based security solutions, privacy risks, and management strategies for large-scale p2p networks. The research also discusses the integration of blockchain technology in p2p networks, and the effectiveness of incentive mechanisms in promoting cooperation and resource sharing. The study also examines the management challenges of large-scale p2p networks and proposes centralized management frameworks, distributed monitoring systems, and dynamic resource allocation algorithms to improve system performance and reliability. The research also discusses the integration of incentive mechanisms with blockchain technology and the effectiveness of different content replication strategies in improving performance and reliability in p2p networks.The proposed system is a mobile file sharing application designed to facilitate safe and efficient file transfers. Leveraging both barcode and pairing code mechanisms, this system ensures a seamless and user-friendly process for connecting devices and sharing files.

Materials and Methods

The software development methodology adopted for the study is the incremental model. The type and source of data are primary and secondary. 3.1 Analysis of the Proposed System

Applications for file sharing that use pairing codes and barcode scanning provide notable gains in efficiency, security, and user experience. Barcode scanning is a quick and easy way to connect, which makes it perfect for short device transfers. When barcode scanning is not an option, pairing codes provide a dependable substitute, although being a little more complicated. Both approaches address important issues with file-sharing apps, like usability, security, and cross-platform compatibility, which makes them useful components of contemporary file-sharing programs.3.1.1 Use Case Diagram

A use case diagram visually illustrates interactions between users and a system, including registration, login, file search, uploading, downloading, managing files, sharing,

viewing transfer status, and connecting to peers. The use case diagram is depicted in figure 3.1.

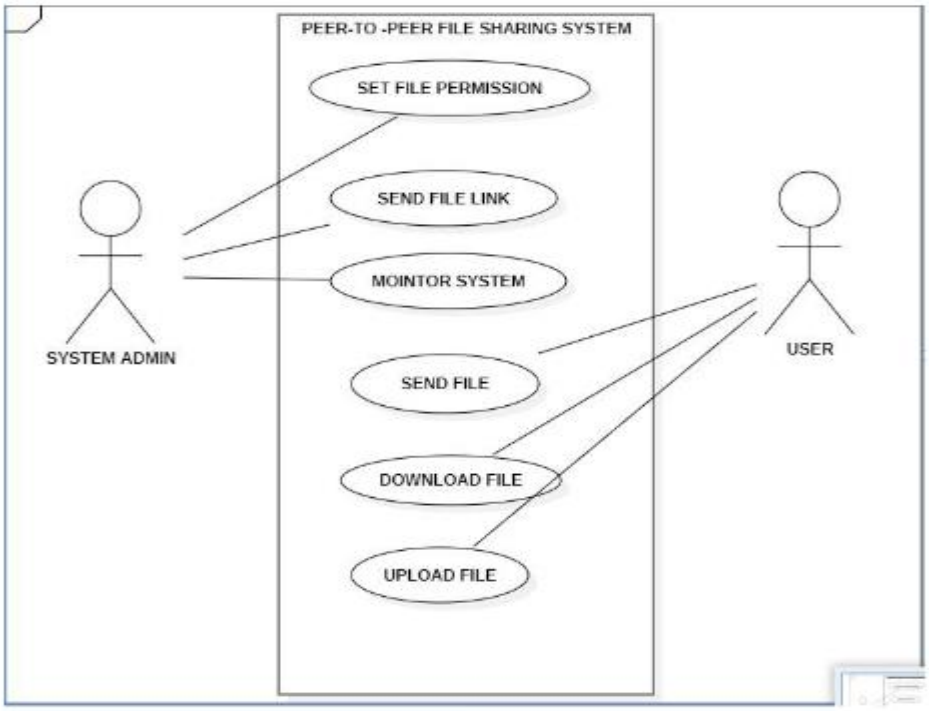


Figure 3.1 : Use Case Diagram for File Sharing Application

3.1.2 Activity Diagram

An activity diagram outlines the flow of activities in the system, demonstrating the sequence of actions taken by the users and the system. The activity diagram is depicted in figure 3.2

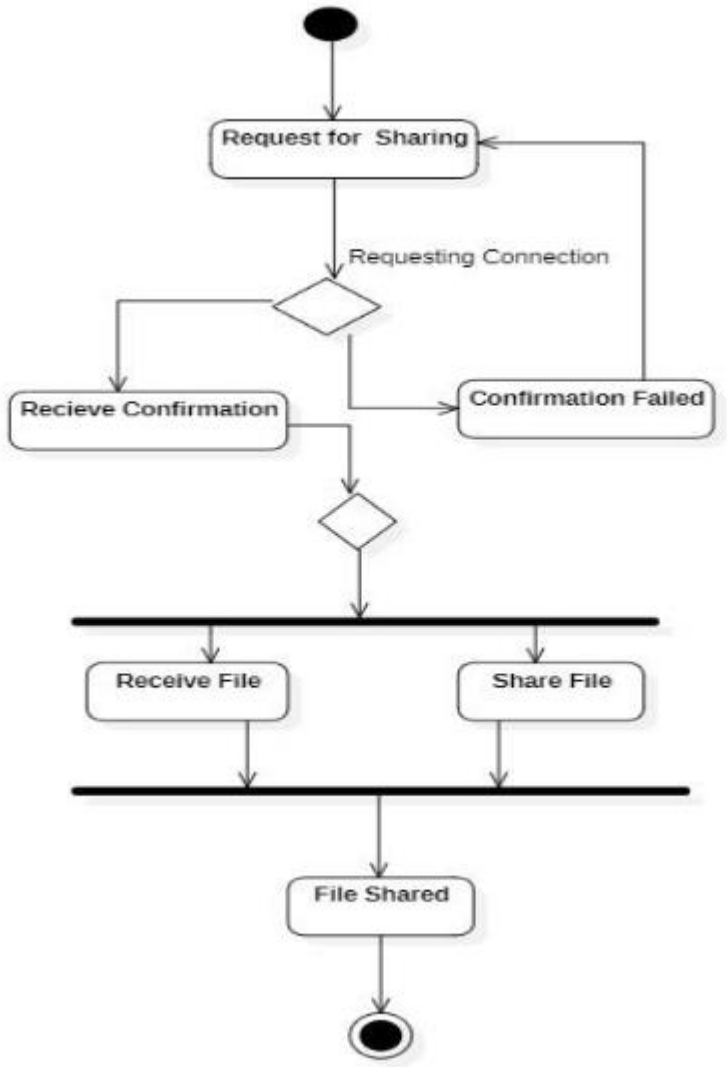


Figure 3.2 Activity Diagram for file Sharing Application

3.1.3 Architectural Design

The P2P file sharing system's architectural design comprises multiple layers and components, ensuring a seamless and efficient filesharing experience. This architectural design is depicted in figure 3.3

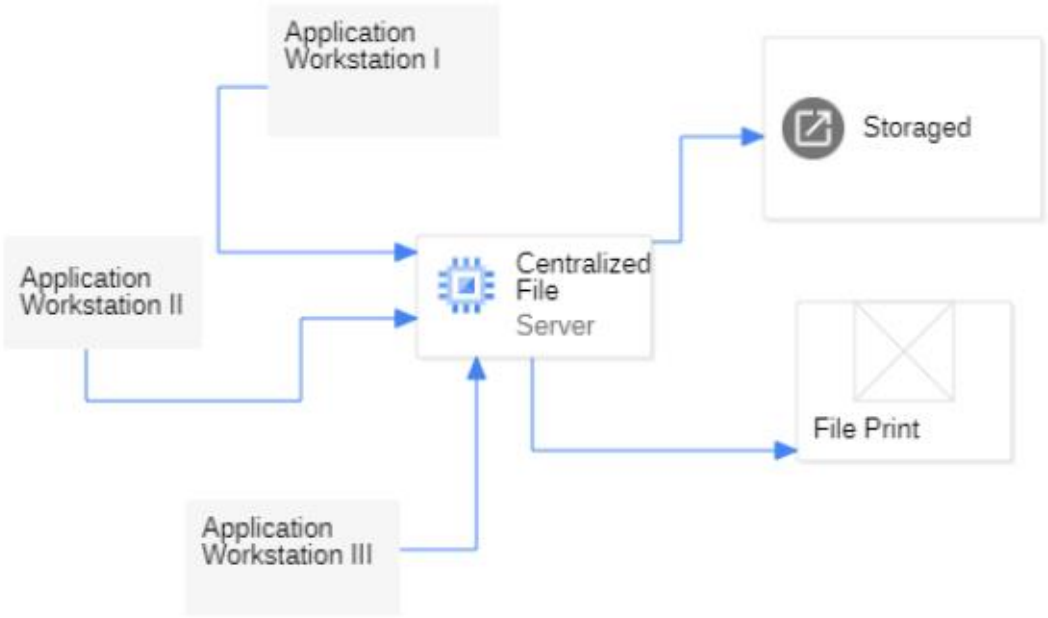


Figure 3.3: Architectural Design for File Sharing Application

3.1.4 Input Layout

The Input Layout design is depicted in figure 3.4

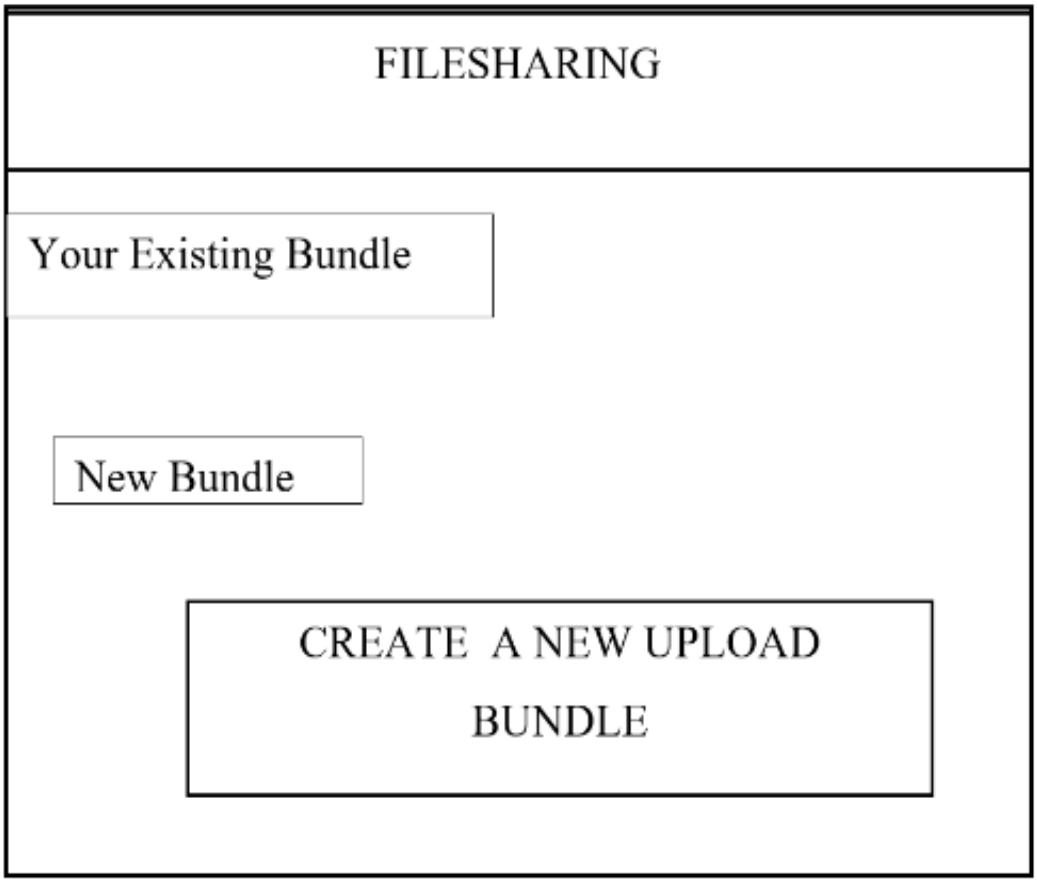


Figure 3.4 Input layout design for peer-to-peer file sharing application

3.1.5 Program Flowchart

The Flowchart design is depicted in figure 3.5

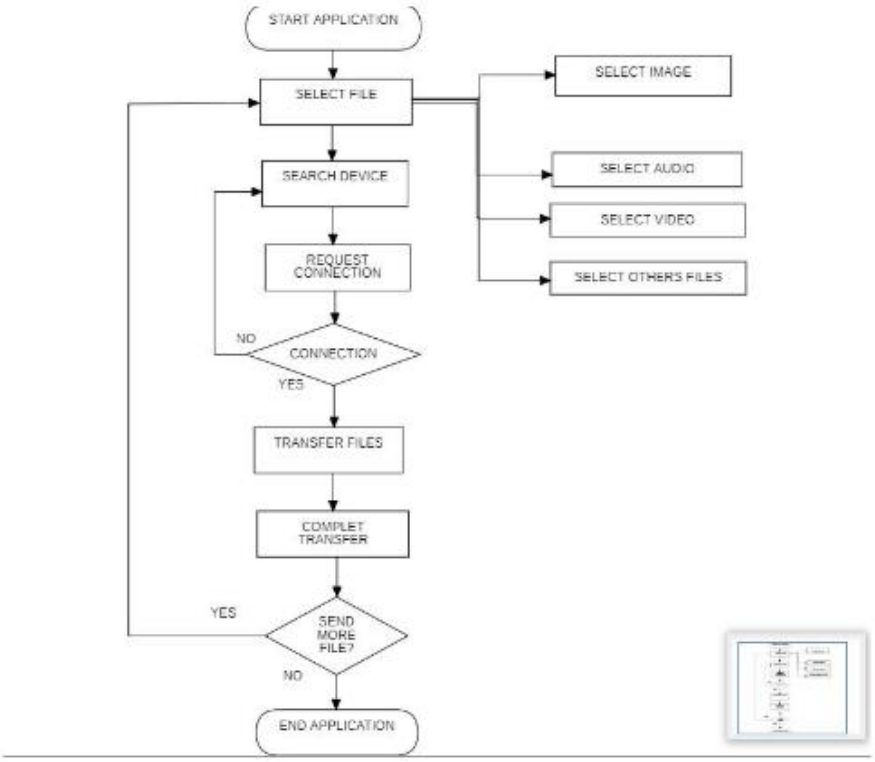


Figure3.5 Flowchart Design for File Sharing Application

3.1.6 Output Format

This Output Format design is depicted in figure 3.6Figure 3.6 Output Format Design for File Sharing Application

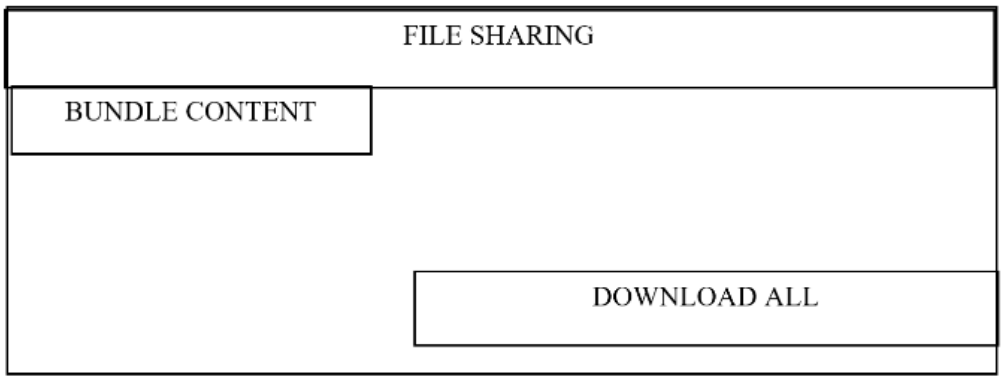


Figure 3.6 Output Format Design for File Sharing Application

Results and Discussion

4.1 Choice of Programming LanguageChoosing the appropriate programming languages for a mobile file sharing application involves considering the needs for both front-end and back-end development. In this system, JavaScript will be used for the mobile

application front-end, while PHP will be utilized for back-end services. This combination leverages the strengths of both languages to create a robust, efficient, and user-friendly file sharing application.The suggested system is broken down into separate yet connected modules, each having a defined set of duties. The system's scalability, maintainability, and security are guaranteed by its modular architecture. PHP offers strong server-side capability, whereas JavaScript, with its extensive ecosystem and cross-platform capabilities, is employed for the client-side. When combined, these technologies produce a feature-rich and effective mobile file-sharing program.4.2 Program Output The program output is show in the figures that follows.

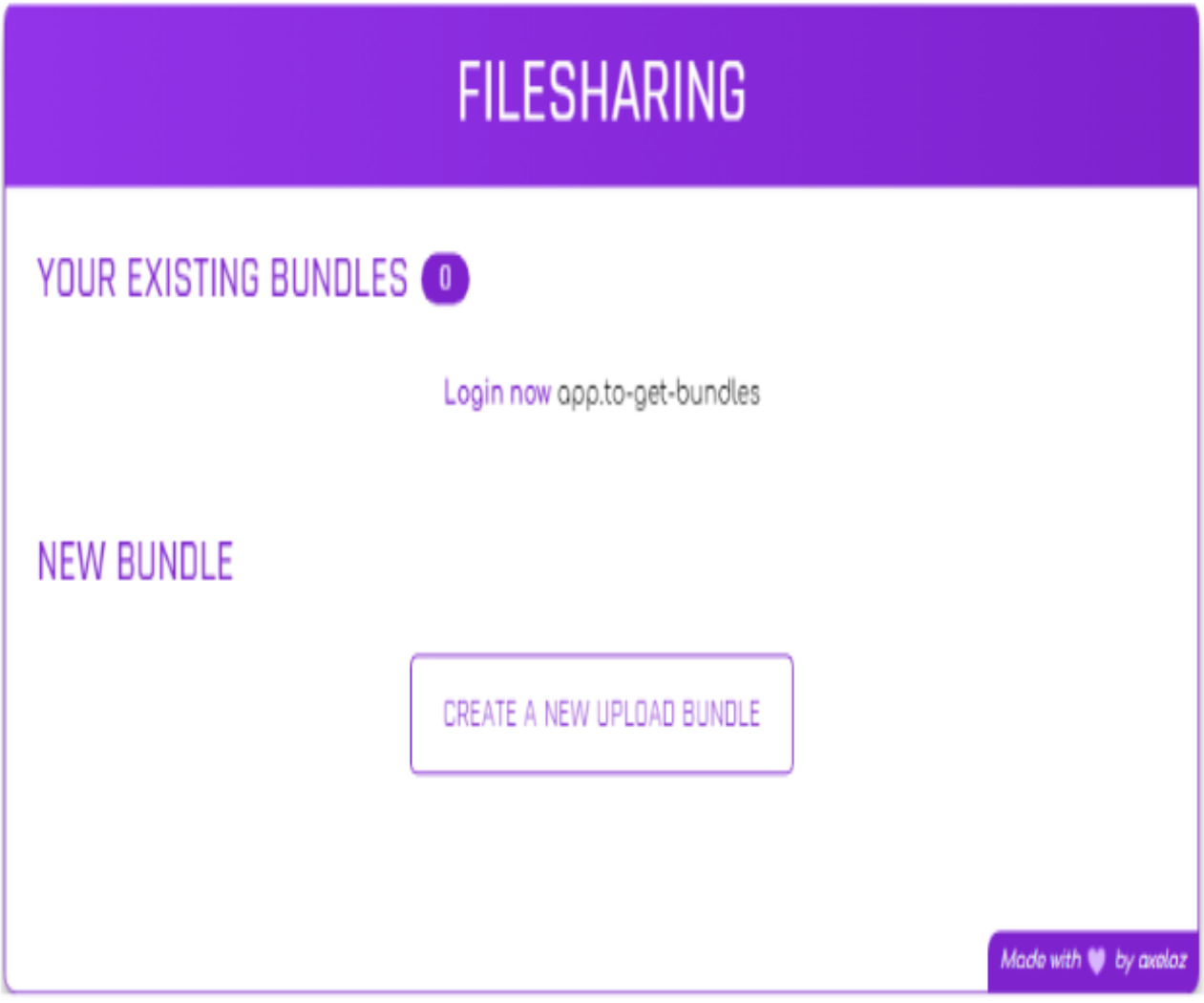


Figure 4.5 Create New Upload Bundle Layout

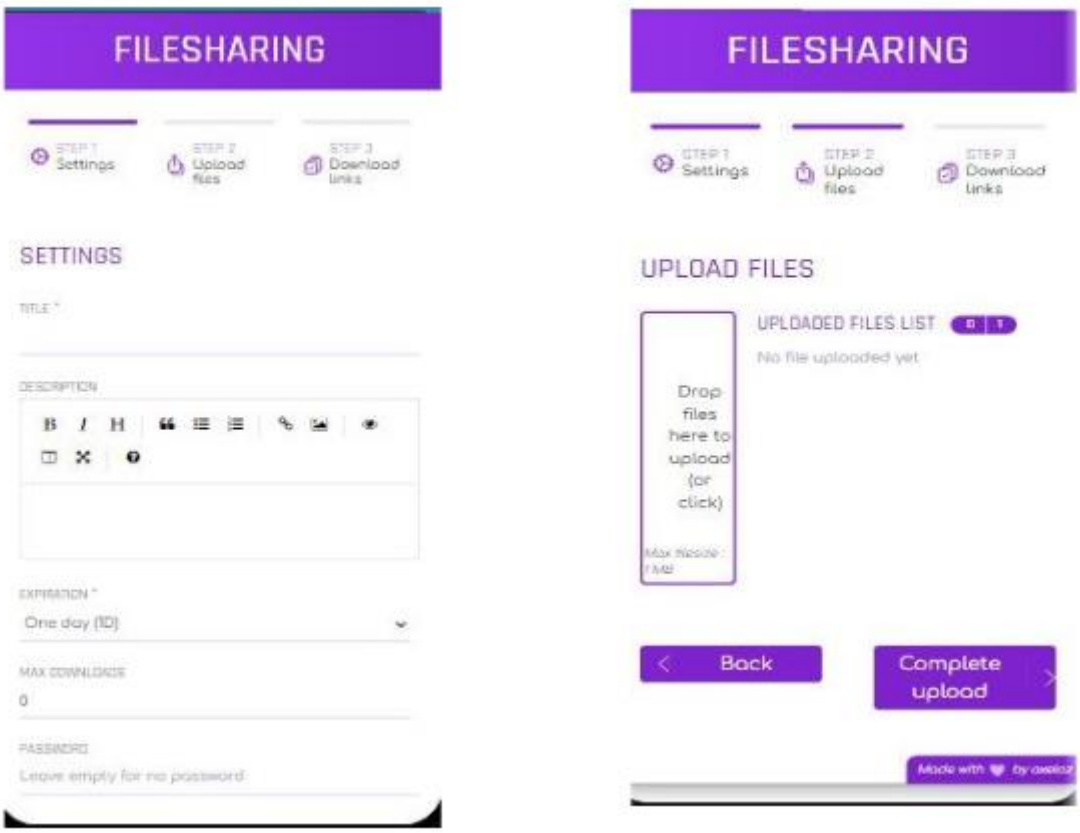


Figure 4.6 Description Layout and Upload File

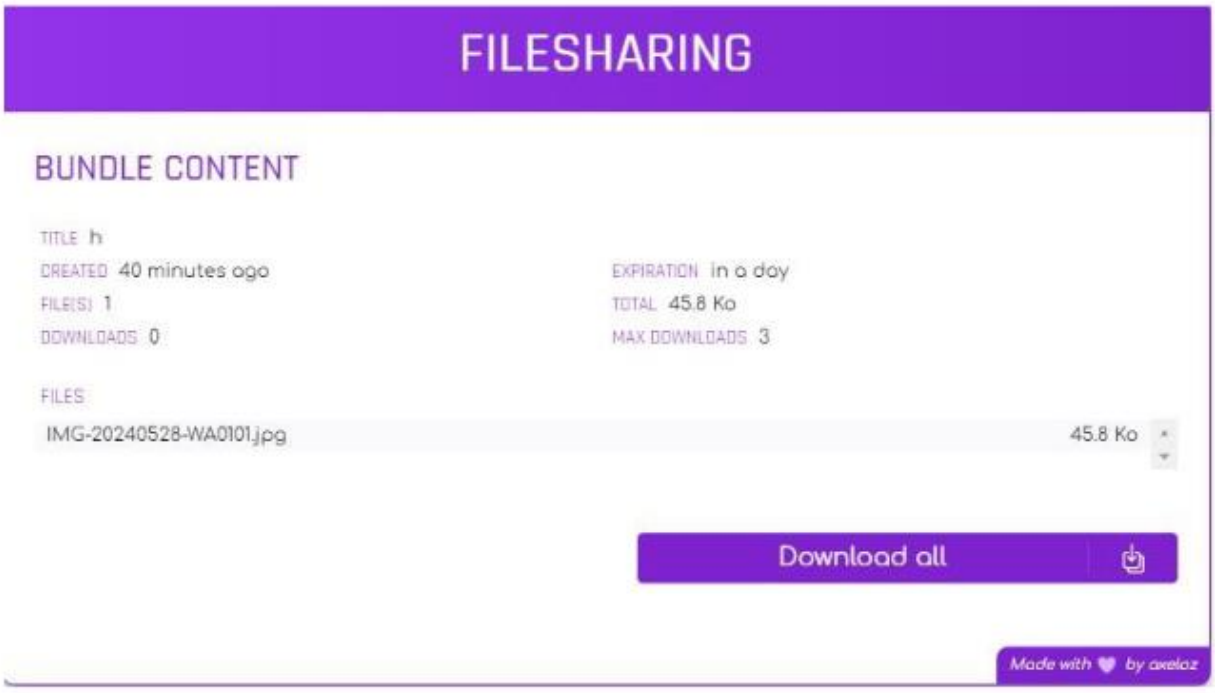


Figure 4.7 Download Screen

The development of a file-sharing application that combines barcode scanning and a manual

pairing code system addresses limitations in current file-sharing solutions. This application enhances accessibility, reliability, and user convenience by offering an alternative to barcode-only methods, ensuring seamless file transfers across various devices and environments.

The integration of dual methods for initiating file transfers mitigates issues related to hardware constraints and environmental factors, promoting greater inclusivity for users with diverse needs and capabilities. This approach sets a new standard for user-friendly and flexible file-sharing solutions, paving the way for more robust and versatile applications in the future.

Careful attention was given to technical, security, usability, and deployment constraints throughout the study, resultingin an innovative and reliable file-sharing application that meets the evolving demands of users while maintaining high standards of security and efficiency.

This study contributes significantly to the field of digital communication and data sharing by providing a practical and effective solution for everyday use, overcoming the challenges associated with existing systems. The successful implementation of this application demonstrates the potential for future advancements and improvements in file-sharing technologies, ultimately enhancing the digital experience for users worldwide

Conclusion

Based on the findings and conclusions of this study, the following recommendations are provided for future enhancements and improvements:

I. Enhance Security and Privacy: Implement robust security measures, such as end-to-end encryption, secure peer authentication, and content access control, to ensure the privacy and protection of user data and shared files.
II. Improve Content Discovery: Explore advanced content indexing and search algorithms, leveraging machine learning techniques, to provide more intelligent and personalized content discovery for users.
III. Integrate with Cloud Storage: Investigate the integration of cloud storage services, allowing users to seamlessly access and share files stored in their personal cloud accounts through the P2P file sharing application.
IV. Extend to Multimedia Sharing: Expand the application's capabilities to support the sharing and streaming of multimedia content, such as videos and music, among mobile devices.
V. Enhance Cross-Platform Compatibility: Ensure the application's compatibility and seamless integration across different mobile platforms, including Android,

iOS, and potentially other emerging mobile ecosystems.

VI. Conduct Extensive User Testing and Feedback: Engage in extensive user testing and feedback collection to continually improve the application's user interface, usability, and overall user experience.

REFERENCES

Adya, A., Bolosky, W. J., Castro, M., Cermak, G., Chaiken, R., Douceur, J. R., Howell, J., Lorch, J. R., Theimer, M., & Wattenhofer, R. P. (2002). Farsite: Federated, available.

Anderson, R. (2020). Peer-to-peer file sharing: A study on BitTorrent protocol. Journal of Network Protocols, 34(2), 112-128.

Ari, I., & Miller, E. L. (2004). ICPADS -Caching support for push-pull data dissemination using data-snooping routers. Proceedings of the 2004 International Conference on Parallel and Distributed Systems, 101.

Carlsson, N. (2006). Scalable downloadprotocols. Proceedings of the International Conference on Paralleland Distributed Systems.

Choi, Y., Park, S., & Lee, K. (2022). Google Nearby Share: Facilitating proximity-based mobile interactions. Proceedings of the 2022 International Conference on Mobile Computing, 315-327.

Gu, X., & Nahrstedt, K. (2006). Distributed multimedia service composition with statistical QoS assurances. IEEE Transactions on Multimedia, 8(1), 141. https://doi.org/10.1109/tmm.2005.861284.

Gunawardena, C. N., & McIsaac, M. S. (2004). Distance education. In A handbook of research in educational communications and technology (2nd ed., pp. 355-395). University of New Mexico and Arizona StateUniversity.

Gupta, S., Kumar, A., & Verma, R. (2019). Efficiency analysis of Wi-Fi Direct in mobile file sharing. International Journal of Mobile Computing, 27(3), 145-159.

Hasan, R., Anwar, Z., Yurcik, W., Brumbaugh, L., & Campbell, R. (2005). A survey of peer-to-peer storage techniques for distributed file systems. International Conference on Information Technology: Coding and Computing (ITCC'05) -Volume II, 205-213. https://doi.org/10.1109/ITCC.2005.42.

Huang, G., & Chen, Z. (2012). A optimal scheduling strategy for data-driven peer-to-peer streaming. Proceedings of the International Conference on Computer and Information Science, 165. https://doi.org/10.1007/978-3-642-30454-5_12.

Hvid, H. (2019). Jonas Axelsson, Jan Ch. Karlsson, Egil J. Skorstad (2019): Collective mobilization in changing conditions. Worker collectivity in a turbulent age.Nordic Journal of Working Life Studies. https://doi.org/10.18291/njwls.v9i4.117781.

Lee, J., & Kim, H. (2020). AirDrop: Secure file sharing within Apple's ecosystem. International Journal of Information Security, 42(4), 210-223.

Moore, L. J., Dickson-Deane, C., & Galyen, K. (2011). eLearning. Internet and Higher Education, 14(2), 129-135.

Rahman, M., Singh, P., & Kaur, S. (2021). Security and performance challenges in P2P mobile applications: The case of Xender. Journal of Mobile Technology, 29(1), 89-104.

Rountree, D. (2010). Network security. In Security for Microsoft Windows system administrators (pp. 71-107). https://doi.org/10.1016/B978-1-59749-594-3.00003-X

Saroiu, S., Gummadi, P. K., & Gribble, S. D. (2002). Measurement study of peer-to-peer file sharing systems. Proc. SPIE 4673, Multimedia Computing and Networking 2002.

Smith, A., & Jones, B. (2018). The role of Bluetooth in mobile file sharing. Wireless Communications Review, 31(2), 67-82.

Tang, Y., Sun, L. F., & Zhang, K. Y. (2007). ICME -Longer, Better: On extending user online duration to improve quality of streaming service in P2P networks. Proceedings of the 2007