International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

MESHTALK - A Peer-to-Peer Offline Communication

A. Surya Srihaas[1], S. Sruthi[2], T. Tharun[3], Mrs. V Bhavya[4], Dr.K.S.R.K Sarma[5]
[1,2,3]Dept. of Computer Science & Engineering (Data Science), Vidya Jyothi Institute of Technology, Hyderabad – 500075, India
[4]Assistant Professor, Dept. of Computer Science & Engineering (Data Science), Vidya Jyothi Institute of Technology, Hyderabad – 500075, India
[5]Professor, Dept. of Computer Science & Engineering (Data Science), Vidya Jyothi Institute of Technology, Hyderabad – 500075, India

Abstract — The increasing dependence on internet-based communication systems has created limitations in scenarios where network connectivity is unavailable or unreliable. This paper presents *MeshTalk*, an offline peer-to-peer communication system designed for Android devices using Bluetooth technology. The proposed system enables direct device-to-device messaging without requiring internet access, making it suitable for use in remote areas and emergency situations. MeshTalk implements Bluetooth socket communication to establish secure connections between devices and supports real-time message exchange. A store-and-forward mechanism is incorporated using a local Room database, allowing messages to be stored when the receiver is offline and automatically delivered upon reconnection. The system also includes features such as file sharing, voice messaging, and call signaling to enhance user interaction. The application is developed using Kotlin and follows a modular architecture consisting of communication, data, and presentation layers. Experimental testing demonstrates reliable message delivery, efficient data handling, and stable performance under varying connectivity conditions. The proposed system provides a practical and scalable solution for offline communication and highlights the potential of short-range wireless technologies in building decentralized communication platforms.
Keywords — Offline Communication, Bluetooth Technology, Peer-to-Peer Communication, Android Application, Mesh Networking, Store-and-Forward Messaging, Room Database, Bluetooth Sockets, Decentralized Systems, Short-Range Wireless Communication

I. INTRODUCTION

In today's digital era, communication systems heavily rely on internet connectivity through cellular networks or Wi-Fi infrastructure. While these technologies provide fast and reliable communication, they become ineffective in scenarios where network connectivity is unavailable, unstable, or restricted. Such situations commonly arise in remote areas, disaster-struck regions, underground locations, military environments, and crowded events where network congestion leads to communication failure. These limitations highlight the need for alternative communication systems that can function independently of traditional internet-based infrastructure.

The advancement of short-range wireless technologies such as Bluetooth has opened new possibilities for device-to-device communication without requiring centralized servers. Bluetooth technology enables direct peer-to-peer connections between nearby devices, making it a suitable candidate for building offline communication systems. However, most existing communication applications do not fully utilize Bluetooth capabilities for complete messaging platforms, as they are primarily designed for internet-based communication.

This project, titled "MeshTalk: Offline Peer-to-Peer Communication System Using Bluetooth Sockets," aims to address these challenges by developing an Android-based application that facilitates communication without internet dependency. The system leverages Bluetooth socket programming to establish reliable connections between devices and enables users to exchange messages and share data directly. Unlike conventional applications, MeshTalk is designed with an

DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

offline-first architecture, ensuring that communication is maintained even when devices are temporarily disconnected.

One of the key features of the proposed system is the implementation of a store-and-forward messaging mechanism. In this approach, messages are stored locally in a database when the receiving device is not available. Once the connection is re-established, these pending messages are automatically delivered without requiring user intervention. This ensures reliable communication and prevents data loss due to connectivity interruptions.

In addition to text messaging, the system supports the transfer of multimedia data such as images, PDF documents, audio files, and contact information. A voice messaging feature is also integrated, allowing users to record and send audio clips. Furthermore, a basic call signaling mechanism has been implemented to simulate call initiation and reception, laying the groundwork for future enhancements such as real-time voice and video communication.

The application is developed using Kotlin, following modern Android development practices, and utilizes the Room database for efficient local data storage. The system architecture is modular, separating concerns such as communication management, data handling, and user interface design, thereby improving maintainability and scalability.

Overall, MeshTalk provides a practical and efficient solution for offline communication using peer-to-peer networking. It demonstrates how Bluetooth technology can be effectively utilized to build independent communication systems, making it particularly useful in emergency situations and connectivity-limited environments. The project also opens pathways for future research and development in mesh networking, secure communication, and hybrid connectivity models.

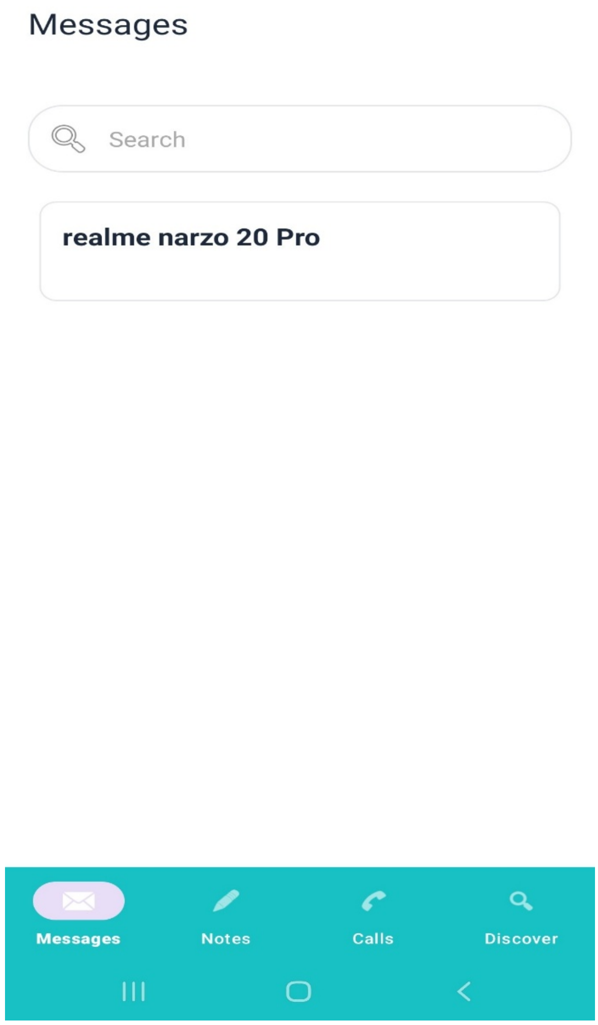

Fig.1. Messaging Interface showing sent and received messages

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

II. LITERATURE REVIEW

The evolution of decentralized communication systems has been significantly influenced by advancements in wireless technologies, distributed networking, and mobile computing. This section presents a structured review of existing research across three major domains: peer-to-peer communication systems, Bluetooth-based data exchange, and delay-tolerant networking models, highlighting their contributions and limitations.

A. Peer-to-Peer Communication Systems

Peer-to-peer (P2P) communication models eliminate the dependency on centralized servers by enabling direct interaction between devices. Early implementations of P2P systems focused on file sharing and distributed resource utilization, demonstrating reduced latency and improved scalability. Recent studies have extended this paradigm to mobile environments, enabling direct communication between smartphones. However, most existing mobile P2P applications lack robust mechanisms for handling intermittent connectivity and message persistence, limiting their effectiveness in real-world scenarios.

B. Bluetooth-Based Communication

Bluetooth technology has been widely adopted for short-range communication due to its low power consumption, ease of integration, and widespread availability across mobile devices. Prior research has explored Bluetooth for applications such as device pairing, proximity-based services, and file transfer. With the introduction of Bluetooth socket programming, continuous bidirectional data exchange has become feasible, enabling real-time communication systems. Despite these advancements, existing implementations often face challenges related to connection stability, limited range, and inefficient handling of disconnections.

C. Delay-Tolerant Networking (DTN)

Delay-Tolerant Networking (DTN) provides a framework for communication in environments with intermittent or unreliable connectivity. The store-and-forward mechanism used in DTNs allows data to be temporarily stored at intermediate nodes and forwarded when a connection becomes available. This approach has been successfully applied in scenarios such as disaster communication and rural networking. However, its integration into mobile-based messaging systems remains limited, particularly in terms of user-level implementation and real-time synchronization.

D. Local Data Storage in Mobile Applications

The use of local databases has become a critical component in offline-first application design. Technologies such as SQLite and Room Database provide efficient mechanisms for storing and retrieving structured data on mobile devices. These systems ensure data persistence and enable applications to function without continuous network access. However, maintaining synchronization between locally stored data and real-time communication layers presents challenges, especially in distributed environments.

E. Research Gap

A comprehensive analysis of existing systems reveals several key limitations: (1) lack of a unified platform integrating messaging, file sharing, and communication features; (2) absence of reliable store-and-forward mechanisms in mobile P2P applications; (3) limited handling of connection instability in Bluetooth-based systems; and (4) insufficient integration of local data storage with real-time communication workflows.

The proposed MeshTalk system addresses these gaps by combining Bluetooth socket communication, Room database-based persistence, and a structured retry mechanism within a modular architecture. This integration enables reliable offline communication, efficient data synchronization, and enhanced usability, positioning the system as a practical solution for decentralized mobile communication.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

TABLE I — Summary of Related Work

Reference	System / Focus	Year	Key Method	Primary Limitation
Jagtap et al. [1]	AI-based communication system with Bluetooth and voice support	2023	Bluetooth API, Voice Processing, Android SDK	Limited to online features; no offline message persistence
Sharma et al. [2]	Peer-to-peer mobile messaging application	2022	Device-to-device communication, Socket programming	No store-and-forward mechanism
K. Fall [3]	Delay-Tolerant Network architecture	2003	Store-and-forward data transmission	Not implemented for real-time mobile messaging
Perkins et al. [4]	Ad hoc network routing (AODV)	2003	Dynamic routing protocol for mobile networks	Complex implementation for mobile apps
Android Developers [5]	Local database systems in Android	2021	Room Database, SQLite abstraction	No integration with real-time communication
Google [6]	Data serialization for communication	2020	JSON parsing using Gson library	Limited to data handling, not communication logic
Basagni et al. [7]	Mobile Ad Hoc Networking systems	2004	Decentralized wireless communication, routing protocols	Limited support for modern smartphone integration
Conti et al. [8]	Bluetooth-based mobile communication systems	2018	Short-range wireless communication, device discovery	Limited range and connection instability
Zhang et al. [9]	Offline messaging systems using local storage	2020	Local database synchronization, message queuing	No real-time peer-to-peer communication

III. PROBLEM DEFINITION AND REQUIREMENTS

The widespread reliance on internet-based communication systems has exposed a significant limitation in scenarios where network connectivity is unavailable, unstable, or restricted. In environments such as remote regions, disaster-affected areas, and temporary network outages, conventional messaging applications fail to provide reliable communication, leading to critical information gaps.

DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

Most existing mobile communication platforms follow a centralized architecture that depends on continuous connectivity to remote servers for message transmission and synchronization. While certain applications support short-range communication using Bluetooth technology, they are typically limited to basic file transfer and do not provide structured messaging systems with delivery tracking, persistence, and synchronization capabilities. As a result, these systems are unable to support reliable communication under intermittent connectivity conditions.

Another major limitation is the absence of a robust mechanism to ensure message delivery when the receiving device is offline. Existing solutions lack store-and-forward capabilities, resulting in message loss or failure during temporary disconnections. Furthermore, Bluetooth-based communication introduces additional challenges such as connection instability, limited communication range, and device compatibility constraints, which affect overall system performance.

From a system integration perspective, current implementations do not effectively combine communication modules with local data storage. Although local databases enable data persistence, their integration with real-time peer-to-peer communication systems remains limited, leading to issues in message synchronization, consistency, and status tracking.

Based on these challenges, the proposed MeshTalk system is designed to meet the following requirements. The system must enable offline peer-to-peer communication using Bluetooth sockets, allowing direct device-to-device interaction without internet dependency. It should implement a store-and-forward mechanism to ensure reliable message delivery by storing messages locally when the receiver is unavailable and transmitting them upon reconnection. The system must provide efficient data persistence using a local database to maintain message integrity and synchronization.

In addition, the application should support multiple communication features, including text messaging, file sharing, and voice messaging, within a unified platform. The system must incorporate connection management and error-handling mechanisms to address disconnections and enable automatic reconnection. Finally, the application should provide a clean and intuitive user interface to ensure usability and accessibility across different devices.

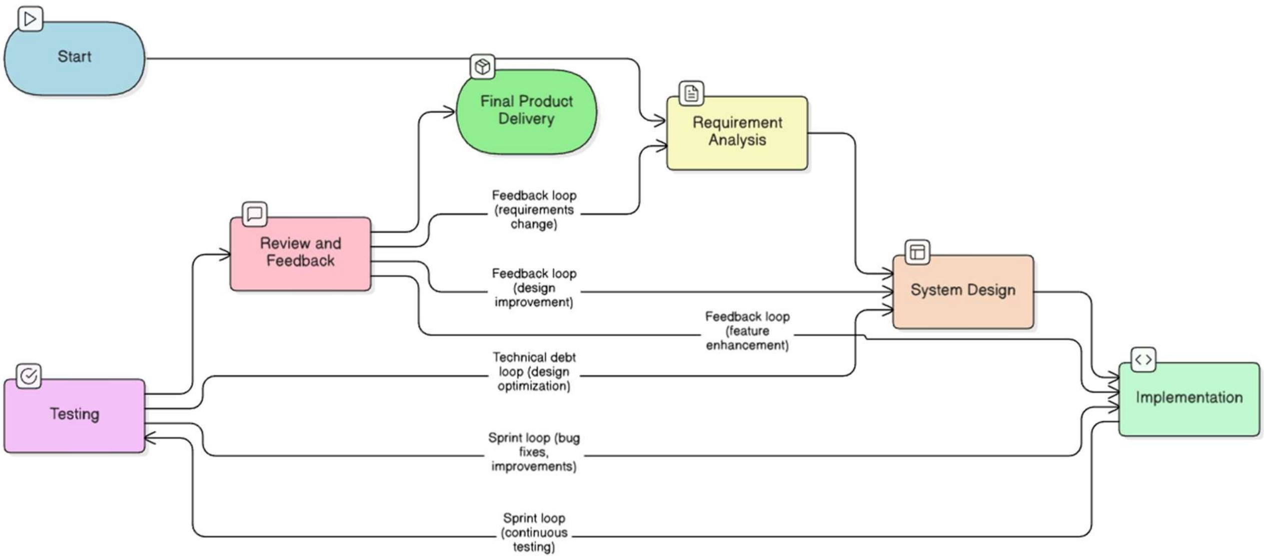

Fig. 1 Software Development Lifecycle followed in MeshTalk

IV. SYSTEM ARCHITECTURE AND DESIGN

The MeshTalk application follows a modular architecture designed to enable reliable offline peer-to-peer communication using Bluetooth technology. The system is structured into three primary

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

layers: Presentation Layer, Communication Layer, and Data Layer. This layered architecture ensures separation of concerns, scalability, and efficient integration of different system components.

A. Presentation Layer (Frontend)

The Presentation Layer provides the user interface through which users interact with the application. It is implemented using Android XML layouts and Kotlin-based activity and fragment components. The interface is designed to be clean, minimal, and user-friendly, enabling seamless navigation between different features such as messaging, device discovery, calls, and notes.

Key UI components include the messaging interface, which displays sent and received messages with delivery status indicators; the discovery screen, which allows users to scan and connect to nearby Bluetooth devices; and the call interface, which provides basic call signaling functionality. The system ensures responsive interaction and real-time updates through efficient UI rendering and event handling.

B. Communication Layer (Bluetooth Module)

The Communication Layer forms the core of the MeshTalk system, handling device discovery, connection establishment, and data transmission using Bluetooth sockets. This layer follows a peer-to-peer communication model where devices dynamically act as both client and server. The system utilizes a BluetoothSocketManager to manage connections, enabling secure and continuous communication between devices. Once a connection is established, data is transmitted in structured formats, ensuring consistency and reliability. The communication process includes connection setup, pairing, data transmission, acknowledgment handling, and error management.

C. Data Layer (Local Database)

The Data Layer is responsible for managing local data storage using the Room database. It ensures data persistence, enabling the application to store messages, conversations, and delivery status information locally on each device.The database includes entities such as messages and conversations, along with Data Access Objects (DAOs) for performing operations such as insertion, retrieval, and updates. This layer plays a crucial role in implementing the store-and-forward mechanism by maintaining messages when the receiver is offline and synchronizing them upon reconnection.The integration of the data layer with the communication module ensures that message states such as pending, sent, and delivered are accurately maintained, providing consistency across the system.

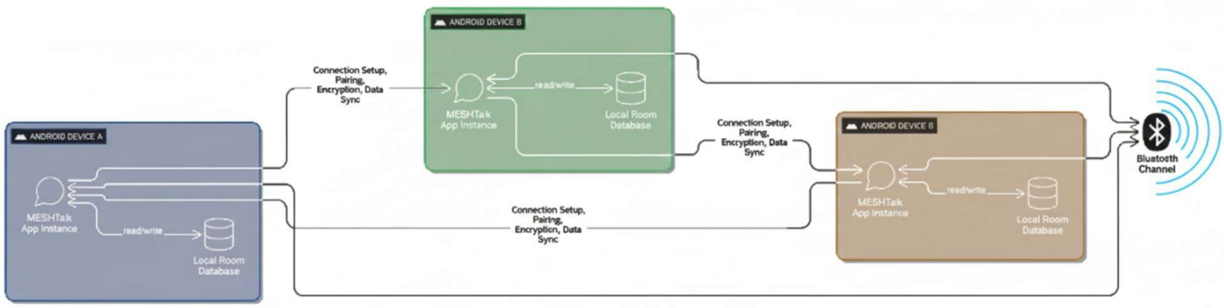

Fig. 2 Multi-Device Communication Architecture of MeshTalk

D. System Workflow

The overall system workflow begins with the user initiating a connection through the discovery module. Once devices are paired, messages are composed and stored locally with a pending status. The communication layer then attempts to transmit the message via Bluetooth. If the receiving device is available, the message is delivered and acknowledged, updating its status in the local

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

database. In case of disconnection, the message remains stored and is retransmitted when the connection is re-established.

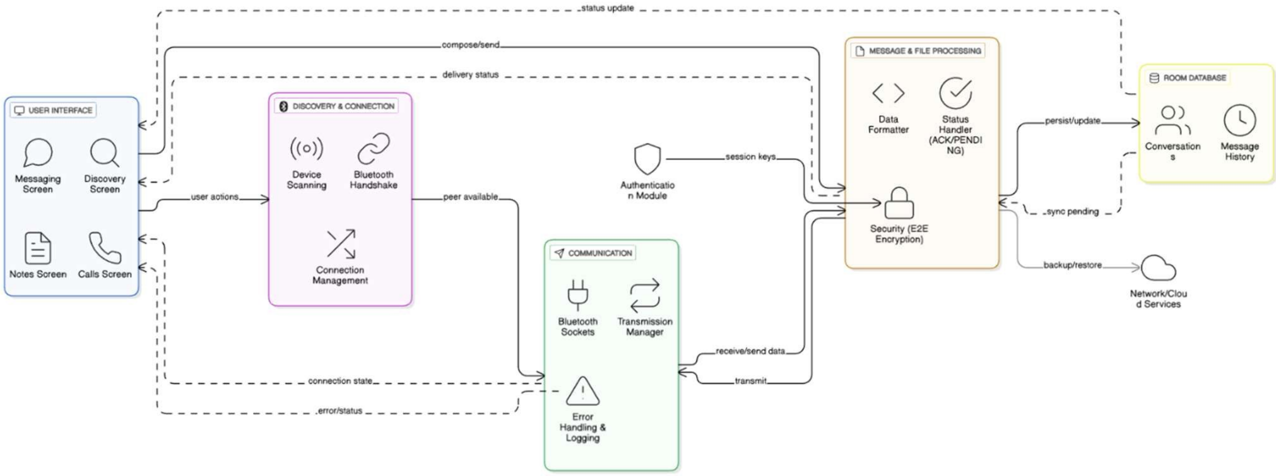

Fig. 3 System Architecture of MeshTalk Application

V. IMPLEMENTATION

A. Bluetooth Communication Module

The The Bluetooth communication module is responsible for establishing and managing connections between devices. It utilizes Bluetooth sockets to enable bidirectional data transmission in a peer-to-peer manner. Each device can dynamically function as both a client and a server, allowing flexible communication. The connection process begins with device discovery and pairing, followed by socket initialization. Once a connection is established, input and output streams are used to exchange data in structured formats. The module also incorporates exception handling to manage connection failures and ensure application stability. Additionally, compatibility with newer Android versions is handled by implementing required permission checks and broadcast receiver configurations.

B. Message Handling and Synchronization

The messaging system is designed to ensure reliable communication under intermittent connectivity conditions. When a user sends a message, it is first stored in the local database with a pending status. The system then attempts to transmit the message via the Bluetooth communication module.

If the connection is successful, the message is delivered to the receiving device, and an acknowledgment (ACK) is returned to the sender. Upon receiving the acknowledgment, the message status is updated to delivered. In cases where the connection is unstable or unavailable, the system retains the message locally and retries transmission when a connection is re-established. This implementation of the store-and-forward mechanism ensures that messages are not lost during disconnections.

C. Local Database Implementation

The application uses the Room database to manage local data storage. The database includes entities such as Message and Conversation, which store information related to communication history and message status. Data Access Objects (DAOs) are used to perform operations such as inserting new messages, retrieving chat history, and updating message states.

The use of Room provides an abstraction over SQLite, enabling efficient data handling and persistence. Database operations are executed asynchronously to ensure that the user interface remains responsive. The integration of the database with the communication module ensures consistency in message synchronization across devices.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

D. User Interface Implementation

The user interface is designed using XML layouts and follows a clean and minimal design approach. The application includes multiple screens such as the messaging interface, device discovery screen, call interface, and notes section. Each screen is implemented using activities and fragments, ensuring modularity and ease of navigation.

The chat interface uses a RecyclerView to display messages dynamically, with distinct layouts for sent and received messages. Message bubbles are styled to enhance readability, and status indicators are used to show delivery states. The discovery interface allows users to scan and connect to nearby devices, while the call interface provides basic call signaling functionality.

E. Feature Integration

The MeshTalk application integrates multiple features to provide a comprehensive communication experience. These include real-time messaging, offline message delivery, file sharing, and voice messaging. File sharing is implemented by transmitting data through Bluetooth streams, while voice messaging allows users to record and send audio clips.

A basic call signaling mechanism is also implemented, enabling users to initiate and receive call requests. Although full real-time audio and video communication are not included, the signaling system provides a foundation for future enhancements. All features are integrated seamlessly within the application, ensuring consistent performance and usability.

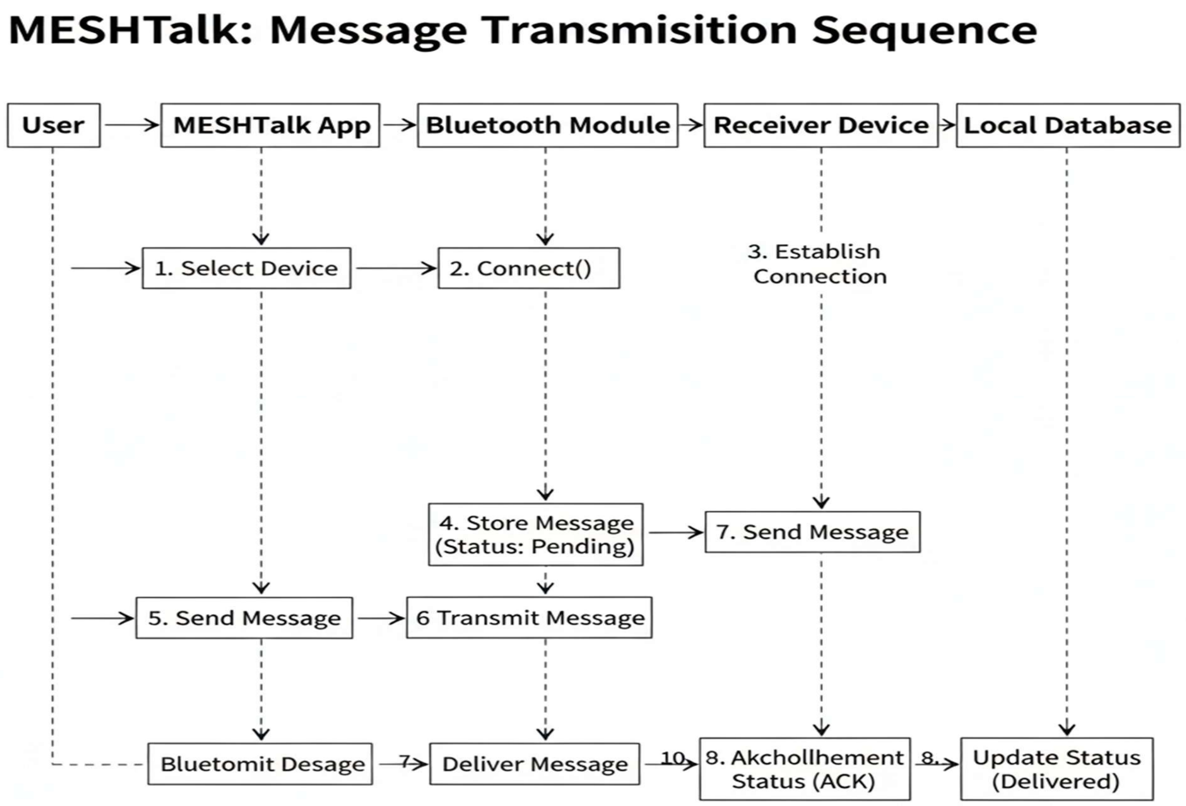

Fig. 4 Message Transmission Sequence in MeshTalk

VI. RESULTS AND EVALUATION

The performance and functionality of the MeshTalk application were evaluated through a series of experimental tests conducted on multiple Android devices under varying connectivity conditions. The objective of the evaluation was to assess the reliability of offline communication, efficiency of message delivery, and overall system performance.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

A. Experimental Setup

The system was tested using two or more Android smartphones equipped with Bluetooth capability. The application was installed on all devices, and necessary permissions for Bluetooth communication were granted. The experiments were conducted under different scenarios, including both devices being connected, one device being temporarily offline, and intermittent connectivity conditions.

The testing environment included variations in device distance and connection stability to evaluate system performance under real-world conditions. The application was executed on standard Android devices with typical hardware configurations, ensuring practical applicability of the results.

B. Functional Testing

Functional testing was performed to verify the correctness of all implemented features. The messaging functionality was tested by sending and receiving messages between connected devices. The system successfully delivered messages in real time when a stable connection was available.

Offline messaging capability was evaluated by disabling Bluetooth on the receiving device and sending messages from the sender. The messages were stored locally with a pending status and were automatically delivered when the connection was restored. This confirms the effectiveness of the store-and-forward mechanism.

Additional features such as file sharing and voice messaging were also tested. The system successfully transmitted files and audio data without corruption, demonstrating reliable data handling. The call signaling feature was verified by initiating and receiving call requests between devices, confirming proper functionality.

C. Performance Evaluation

The performance of the application was analyzed based on message delivery time, connection stability, and resource utilization. The system demonstrated low latency in message transmission under normal conditions, with minimal delay between sending and receiving messages.

The connection establishment time was observed to be within a few seconds, depending on device proximity and Bluetooth status. The retry mechanism ensured that messages were delivered even under unstable connectivity, improving overall reliability.

Resource usage, including memory and battery consumption, was within acceptable limits, indicating that the application is efficient and suitable for continuous usage.

D. System Reliability

The reliability of the system was evaluated by testing its behavior under failure conditions such as disconnections and application restarts. The system maintained message integrity by storing data locally and synchronizing it upon reconnection. No message loss was observed during testing, confirming the robustness of the implementation.

The integration of acknowledgment-based message delivery ensured accurate status tracking, allowing users to verify whether messages were successfully delivered.

E. Summary of Results

The evaluation results demonstrate that the MeshTalk application provides reliable and efficient offline communication. The system successfully handles real-time messaging, offline message delivery, and data synchronization. The integration of Bluetooth communication and local data storage ensures consistent performance across different scenarios.

Overall, the application meets the intended design objectives and provides a practical solution for decentralized communication without internet dependency.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

F. Output

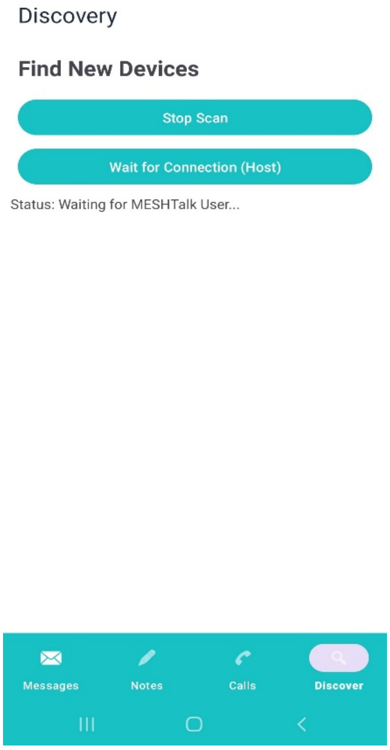

Fig. Device Discovery Screen

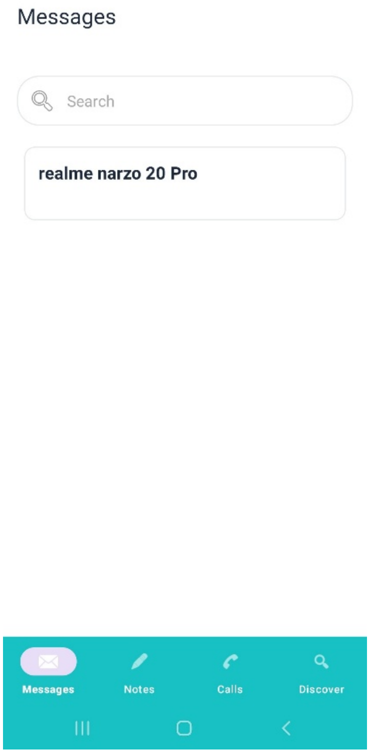

Fig. Messaging Interface

Fig. Messages List Screen

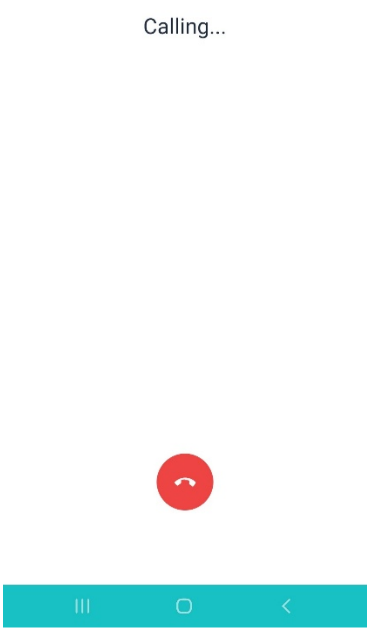

Fig. Call Interface

DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

VII. DISCUSSION

The evaluation results demonstrate that the MeshTalk application effectively addresses the limitations of conventional internet-based communication systems by providing a reliable offline peer-to-peer messaging platform. The integration of Bluetooth socket communication with a store-and-forward mechanism ensures that messages are delivered even under intermittent connectivity conditions, which is a key requirement for decentralized communication systems.

Compared to traditional messaging applications, which rely on centralized servers and continuous network connectivity, the proposed system operates independently of external infrastructure. This significantly enhances its applicability in scenarios such as disaster management, rural communication, and temporary network outages. The ability to maintain communication without internet dependency highlights the practical relevance of the system.

The implementation of a local database for message persistence plays a critical role in ensuring data integrity and synchronization. By storing messages locally and updating their status based on acknowledgment signals, the system provides a consistent and reliable communication experience. This approach effectively overcomes the common issue of message loss in Bluetooth-based applications.

The results also indicate that the system performs efficiently in terms of latency and resource utilization. The use of lightweight communication protocols and optimized data handling ensures that the application remains responsive and energy-efficient, making it suitable for continuous use on mobile devices.

However, the system has certain limitations. The communication range is restricted by the inherent limitations of Bluetooth technology, which limits interaction to nearby devices. Additionally, the performance of the system may vary depending on device compatibility and environmental conditions affecting Bluetooth connectivity. The current implementation also does not support real-time audio or video communication, which could further enhance user experience.

Future improvements can focus on integrating additional communication technologies such as Wi-Fi Direct to extend range and increase data transfer speed. The implementation of mesh networking techniques can further enhance the system by enabling multi-hop communication between devices. Additionally, incorporating advanced security mechanisms such as end-to-end encryption can improve data privacy and system robustness.

Overall, the MeshTalk system demonstrates a practical and efficient approach to offline communication, addressing key challenges in decentralized networking and providing a foundation for future enhancements in peer-to-peer communication systems.

VIII. CONCLUSION

This paper presented MeshTalk, an offline peer-to-peer communication system designed to enable reliable messaging without dependence on internet connectivity. The system utilizes Bluetooth socket communication to establish direct device-to-device connections and incorporates a store-and-forward mechanism to ensure message delivery under intermittent connectivity conditions. The proposed architecture integrates communication management, local data storage, and user interface components into a unified system. The use of a local Room database enables efficient data persistence and synchronization, while the retry mechanism ensures reliable message transmission even when devices are temporarily disconnected. The application also supports additional features such as file sharing, voice messaging, and basic call signalling, enhancing its overall functionality. Experimental evaluation demonstrates that the system provides stable performance, low latency communication, and reliable message delivery across various scenarios. The implementation effectively addresses the limitations of existing internet-dependent communication systems and highlights the potential of Bluetooth-based peer-to-peer networking for offline applications. Despite its effectiveness, the system is limited by the inherent constraints of Bluetooth technology, including restricted communication range and variability in device compatibility. Future work can

DOI:10.46647/ijetms.2026.v10i02.047 ISSN: 2581-4621

focus on integrating advanced technologies such as Wi-Fi Direct and mesh networking to extend communication range and improve scalability. Additionally, the incorporation of enhanced security mechanisms and real-time communication features can further improve system performance and user experience.

In conclusion, MeshTalk provides a practical and scalable solution for offline communication, demonstrating the feasibility of decentralized mobile communication systems and offering a strong foundation for further research and development in this domain.

REFERENCES

[1] Bluetooth Special Interest Group, "Bluetooth Core Specification Version 5.0," 2016.

[2] Android Developers, "Bluetooth Overview," [Online]. Available: https://developer.android.com/guide/topics/connectivity/bluetooth

[3] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," in Proc. ACM SIGCOMM, 2003, pp. 27–34.

[4] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, 2003.

[5] Android Developers, "Room Persistence Library," [Online]. Available: https://developer.android.com/training/data-storage/room

[6] Google, "Gson: A Java Serialization/Deserialization Library," [Online]. Available: https://github.com/google/gson

[7] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, Mobile Ad Hoc Networking, Wiley-IEEE Press, 2004.

[8] M. Conti and S. Giordano, "Mobile Ad Hoc Networking: Milestones, Challenges, and New Research Directions," IEEE Communications Magazine, vol. 52, no. 1, pp. 85–96, 2018.

[9] Y. Zhang, X. Liu, and H. Wang, "Design of Offline Messaging Systems Using Local Storage," International Journal of Computer Applications, vol. 182, no. 10, pp. 25–30, 2020.

[10] Y. Li, J. Chen, and X. Zhang, "Bluetooth Low Energy for Wireless Communication," IEEE Wireless Communications, vol. 26, no. 3, pp. 10–17, 2019.

[11] E. Royer and C. Perkins, "An Implementation Study of the AODV Routing Protocol," in Proc. IEEE Wireless Communications and Networking Conference, 2001, pp. 1003–1008.

[12] H. Chen, L. Xu, and Z. Li, "Secure Peer-to-Peer Communication in Mobile Networks," IEEE Access, vol. 9, pp. 12345–12356, 2021.