International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 10 | Oct 2026 www.irjet.net p-ISSN: 2395-0072

CAREBRIDGE: An Integrated Healthcare Management and Virtual Consultation Platform

Ms.T.Renuka[1], Mr.Ashwith Reddy Nayini[2] Mr.G.P.Dhanush[3], Ms. Ananthula Usha Rani[4]

[1]*Assistant Professor, Dept. of Computer Science and Engineering, HITAM, Telangana, India*
[234]*Student, Dept. of Computer Science and Engineering, HITAM, Telangana, India*

Abstract - *Healthcare institutions coordinate several activities, including patient registration, appointment scheduling, clinical documentation, consultations, prescriptions, and administrative operations. When these activities are handled through disconnected systems, maintaining consistent information and coordinating patient care becomes difficult. Remote consultations introduce additional requirements such as real-time communication, participant authorization, session management, and secure file exchange. CareBridge proposes an integrated platform that combines healthcare workflow coordination with virtual consultation capabilities. Its architecture separates the presentation layer, backend application services, persistent data, and real-time communication infrastructure. The documented technologies include Flutter/Dart, Node.js, Express, TypeScript, Supabase/PostgreSQL, and LiveKit/WebRTC. Additional components support authentication, request validation, database access policies, file handling, and consultation collaboration. The available engineering evidence establishes the presence of relevant modules and test artifacts. However, complete end-to-end integration, security testing, and deployment validation are still required before production readiness can be established. CareBridge therefore provides a modular engineering foundation for healthcare management and remote consultation.*

Keywords: CareBridge, Healthcare Management, Telehealth, Virtual Consultation, Clinical Documentation, Flutter, PostgreSQL, LiveKit, WebRTC, Access Control.

1. INTRODUCTION

Healthcare organizations depend on coordinated information exchange among patients, clinicians, and administrative staff. Activities such as registration, appointment scheduling, consultation management, clinical documentation, prescriptions, and follow-up require consistent information. When these activities operate through separate workflows, they may result in repeated data entry, fragmented records, and manual coordination.

Digital healthcare platforms provide a common environment for organizing these activities. Healthcare management systems maintain structured patient and administrative information, while virtual consultation systems enable remote communication between patients and healthcare professionals. Integrating these capabilities requires consistent user identities, authorization rules, data relationships, and consultation states.

CareBridge is designed as an integrated healthcare management and virtual consultation platform. It connects patient and clinician workflows, appointment-related coordination, clinical documentation, and remote consultation. Its consultation environment is designed to support audio-video communication and supplementary features such as chat, file sharing, whiteboard collaboration, and live captions.

The system separates structured healthcare information from real-time media communication. Application and database services manage persistent records, whereas dedicated communication infrastructure handles audio-video sessions.

1.1 Problem Statement

Healthcare management and virtual consultation involve related but technically different processes. When these processes are not properly connected, the system may experience inconsistent patient identifiers, incomplete consultation documentation, unclear session states, duplicated data entry, and authorization gaps.

The problem addressed by CareBridge is the design of a modular platform that coordinates healthcare administration and remote consultation while maintaining clear boundaries between the user interface, business operations, persistent data, and real-time media services.

1.2 Objectives

The main objectives of CareBridge are:

1. To organize patient, clinician, and appointment-related workflows.
2. To support clinical documentation and consultation records.
3. To provide authorized audio-video consultation capabilities.

4. To support chat, file sharing, and collaborative consultation tools.
5. To establish authentication and role-based authorization.
6. To maintain structured data and controlled file access.
7. To define clear interfaces between frontend, backend, database, and media services.
8. To support systematic functional, integration, and security testing.

1.3 Scope and Contribution

The project scope includes participant information, scheduling-related activities, clinical records, consultation documentation, audio-video sessions, participant access, chat, file sharing, whiteboard collaboration, and captions. Additional areas such as prescriptions, laboratory services, pharmacy, billing, inventory, notifications, and analytics are identified as broader functional scope areas; their implementation and integration must be verified individually.

The main contribution is a unified architectural approach that connects healthcare workflows with remote consultation. It separates persistent clinical information from real-time communication, defines consultation states, and identifies security and integration requirements. The project does not claim a new medical treatment method or proven clinical improvements.

2. LITERATURE REVIEW

Telemedicine has been studied as a healthcare delivery model that extends healthcare services through communication technologies [1]. Dorsey and Topol reviewed the development of telehealth in modern healthcare [2], while Kruse et al. examined factors associated with telehealth and patient satisfaction [3]. These studies provide a foundation for understanding remote consultation and emphasize the importance of usability and communication.

Healthcare information systems organize patient information, clinical documentation, and administrative activities. Their effectiveness depends on consistent data relationships, reliable information handling, and appropriate access restrictions. Therefore, consultation records should be associated with the correct patient, clinician, and appointment context.

Sandhu et al. presented Role-Based Access Control (RBAC), where permissions are associated with roles and users are assigned appropriate roles [4]. This approach is relevant to healthcare applications because patients, clinicians, and administrative personnel require different permissions. Authorization should be enforced through backend services and database operations rather than frontend restrictions alone.

The WebRTC specification defines technologies for real-time media and data communication between compatible endpoints [5]. Successful communication also depends on network conditions, signaling, device permissions, and session configuration.

NIST Special Publication 800-66 Revision 2 provides security guidance related to the HIPAA Security Rule [6]. It serves as a security reference but does not establish that CareBridge is HIPAA-compliant.

The research gap addressed by CareBridge is the practical coordination of healthcare workflows, access control, persistent records, and real-time communication within one modular application. Its contribution is primarily architectural and implementation-oriented.

3. METHODOLOGY

The methodology follows a modular system-engineering approach involving functional requirements, architecture design, data management, consultation lifecycle management, security controls, and verification.

3.1 System Architecture

CareBridge follows a layered architecture consisting of four major layers:

1. Presentation Layer: Provides role-specific screens, forms, healthcare workflows, and consultation controls.
2. Application Layer: Handles business operations, request validation, authentication, authorization, and consultation-related actions.
3. Persistence Layer: Stores structured records and file-related information through database and storage services.
4. Real-Time Communication Layer: Supports audio-video sessions and associated real-time interactions.

This separation allows individual components to be evaluated independently while maintaining defined interfaces between them.

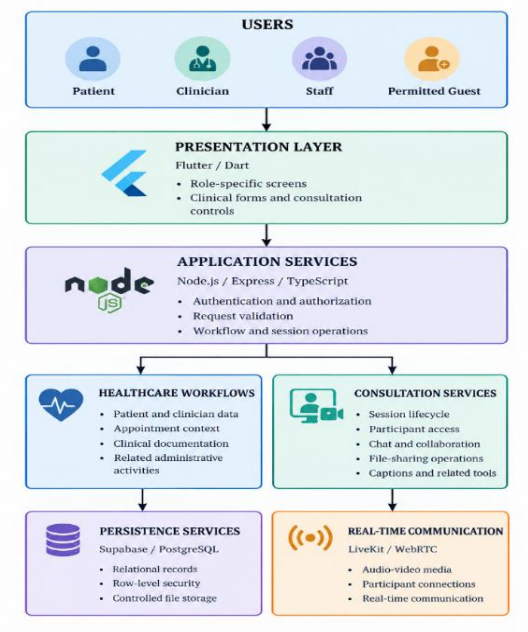


Figure 1. Conceptual Architecture of CareBridge

3.2 Technology Stack

The documented consultation implementation uses the following technologies.

Table -1:

Component	Technology	Responsibility
Frontend	Flutter, Dart	Cross-platform interface
State Management	Riverpod	Application state
Navigation	GoRouter, GetIt	Routing and dependencies
Backend	Node.js, Express, TypeScript	APIs and business logic
Validation	Zod	Request validation
Authentication	JWT-related modules	Authenticated access
Database	Supabase, PostgreSQL	Structured data storage
Database Security	Row-Level Security	Access restrictions
Communication	LiveKit, WebRTC	Audio-video sessions
Supplementary Tools	Web Speech API, MediaPipe-related components	Captions and image processing

These technologies represent the documented consultation stack and should not be interpreted as proof that every wider healthcare-management function is integrated.

3.3 Functional Requirements

The main functional requirements include:

- User authentication and role-aware access.
- Patient, clinician, and appointment-related workflow management.
- Consultation lifecycle management.
- Authorized audio-video communication.
- Clinical documentation and consultation records.
- Controlled file upload and sharing.
- Chat and collaboration features.
- Supported caption and image-processing functions.
- Backend request validation and authorization.

Non-functional requirements include security, privacy, data integrity, reliability, performance, maintainability, compatibility, accessibility, and recoverability. These are design requirements that must be evaluated through appropriate testing.

3.4 Consultation Lifecycle

The consultation lifecycle defines how a session progresses through different states. It helps control admission, participant access, active communication, and completion.



Figure 2. Consultation Lifecycle

A consultation may also be cancelled according to the defined lifecycle rules. The authoritative application service must enforce valid state transitions; frontend labels alone are insufficient.

3.5 Data Management and Security

The system depends on consistent relationships among users, patients, clinicians, appointments, consultations, clinical notes, vital measurements, prescriptions, file metadata, and guest invitations.

Supabase/PostgreSQL supports structured data storage, while Row-Level Security (RLS) provides a mechanism for restricting database operations. Its effectiveness depends on correct policy configuration and deployment.

File storage requires separate access controls. Uploaded documents may be represented through database metadata, while actual files are maintained by storage services. Signed-link expiry and unauthorized access must be tested.

Security controls include authentication, role-based authorization, request validation, data isolation, private-file protection, guest-invitation restrictions, secret management, and consultation lifecycle enforcement. Verification should include expired tokens, unauthorized record access, invalid invitations, and prohibited session transitions.

4. IMPLEMENTATION

The documented implementation combines a Flutter/Dart client, Node.js/Express/TypeScript backend services, Supabase/PostgreSQL persistence, and LiveKit/WebRTC-related communication components.

4.1 Frontend Implementation

Flutter and Dart are used for the client application. The frontend provides the interface through which users access healthcare workflows and consultation functions. Riverpod supports application state management, while GoRouter and GetIt are identified for navigation and dependency access.

The consultation interface contains components for joining sessions, managing audio-video devices, communicating with participants, and using supplementary tools. Browser-dependent functions, including speech-related capabilities, require compatibility testing across supported devices and browsers.

4.2 Backend Services

The backend uses Node.js, Express, and TypeScript. It is responsible for receiving requests, validating inputs, applying authorization rules, and coordinating operations involving database and consultation services.

Zod is identified as a request-validation technology, while token-based authentication and authorization middleware are also documented. Sensitive operations must be enforced by backend services rather than relying only on frontend navigation restrictions.

4.3 Database and File Services

Supabase and PostgreSQL provide the documented persistence services. Row-Level Security definitions provide a mechanism for restricting data access. Database migrations must be applied consistently to avoid differences between the expected and deployed schema.

File-handling functionality includes upload and controlled-access mechanisms. Tests must confirm that unauthorized users cannot retrieve private documents and that temporary access links follow configured expiry rules.

4.4 Real-Time Consultation

LiveKit and WebRTC-related components support audio-video consultation. The media layer manages communication, whereas application services manage participant identity, authorization, session information, and related clinical operations.

Additional documented consultation components include chat, file sharing, whiteboard collaboration, captions, and image-processing functions. These features require individual testing for multiple participants, synchronization, browser compatibility, language support, and device permissions.

4.5 Integration Requirements

The unified system depends on consistent integration between healthcare workflows and consultation services. Important requirements include:

- Patient identity must remain consistent across all related records.
- Appointments must map to the correct consultation sessions.
- Consultation states must remain consistent across the client, backend, and database.
- Clinical documentation must be associated with the correct patient and consultation.
- Guest access must follow admission and authorization requirements.
- File permissions must be consistent across application and storage services.
- Session completion must prevent prohibited re-entry.

5. RESULTS AND DISCUSSION

The available engineering evidence supports the presence of a modular architecture, documented source components, and test artifacts. However, it does not establish that every feature has passed complete end-to-end testing or that the platform is production-ready.

Source files demonstrate that modules exist, while test files demonstrate that verification scenarios have been prepared. Successful test execution provides evidence only for the specific cases and environment tested.

5.1 Functional Coverage

Functional Area	Documented Evidence	Verification Required
Frontend	Flutter/Dart application modules	Complete workflow testing
Backend	Node.js, Express, TypeScript	Deployed API verification
Authentication	Token-related modules and middleware	Authorization testing
Consultation	Lifecycle and media components	End-to-end session testing
Clinical Records	Related data structures and modules	Data integrity and access testing
File Sharing	Upload and access mechanisms	Storage security testing
Collaboration	Chat and whiteboard modules	Multi-participant testing
Captions	Browser-related components	Language and compatibility testing
Image Processing	Image-processing modules	Functionality and accuracy evaluation

5.2 Testing and Verification

The project documentation identifies Flutter and backend test artifacts across different functional areas. However, a fresh complete test run across the entire stack was not established during the reviewed source evaluation.

Backend execution encountered test-launcher or package-environment issues, and the Flutter command-line tool was unavailable in that review environment. Therefore, the presence of test files should not be treated as proof that all tests pass.

The following verification activities remain important:

1. Validate authentication and reject invalid or expired tokens.
2. Prevent unauthorized users from accessing restricted records.
3. Verify guest invitation expiry and access restrictions.
4. Test consultation admission, completion, and prohibited re-entry.
5. Verify clinical record relationships and database policies.
6. Test private-file access and signed-link expiry.
7. Evaluate audio-video connectivity and network interruption handling.
8. Verify chat, whiteboard, and caption behavior across supported devices.

5.3 Overall Findings

The engineering review supports the following findings:

- CareBridge has a documented modular architecture for healthcare workflow coordination and virtual consultation.
- The documented technology stack separates client functionality, backend services, persistent data, and real-time communication.
- Authentication, request validation, database policies, and controlled file access provide a foundation for security controls.
- Explicit consultation states provide a basis for managing valid session transitions.
- Supplementary components include chat, whiteboard collaboration, captions, file handling, and image-processing functionality.
- Test artifacts exist, but a complete passing test run across the entire stack was not established.
- Full integration, security enforcement, and deployment readiness require further verification.

The modular architecture supports maintainability when interfaces and data relationships are correctly defined. However, integration testing is essential because separate components may interpret identities, permissions, or consultation states differently.

No quantitative improvement in administrative efficiency, consultation duration, or clinical outcomes is claimed because controlled measurements are not available.

3. CONCLUSIONS

CareBridge presents an integrated healthcare management and virtual consultation platform designed to coordinate administrative workflows, clinical documentation, and remote consultation within one system concept.

The documented technical foundation includes Flutter and Dart for the client, Node.js with Express and TypeScript for backend services, Supabase and PostgreSQL for data persistence, and LiveKit/WebRTC-related components for real-time communication. Additional documented components include authentication, request validation, database access policies, file handling, chat, whiteboard collaboration, captions, and image processing.

The principal engineering strengths are modular architecture, structured data management, explicit consultation states, and consideration of authorization across multiple layers. However, the available evidence does not establish that every healthcare-management and consultation capability has been fully integrated, deployed, and validated end to end.

Complete functional testing, security review, database configuration verification, file-access testing, and deployment validation are necessary before operational readiness can be established. CareBridge therefore provides a structured engineering foundation for integrated healthcare workflow coordination and virtual consultation.

REFERENCES

[1] R. L. Bashshur, T. G. Reardon, and G. W. Shannon, "Telemedicine: A New Health Care Delivery System," Annual Review of Public Health, vol. 21, pp. 613–637, 2000. https://doi.org/10.1146/annurev.publhealth.21.1.613

[2] E. R. Dorsey and E. J. Topol, "State of Telehealth," The New England Journal of Medicine, vol. 375, no. 2, pp. 154–161, 2016. https://doi.org/10.1056/NEJMra1601705

[3] C. S. Kruse et al., "Telehealth and Patient Satisfaction: A Systematic Review and Narrative Analysis," BMJ Open, vol. 7, no. 8, e016242, 2017. https://doi.org/10.1136/bmjopen-2017-016242

[4] R. S. Sandhu et al., "Role-Based Access Control Models," Computer, vol. 29, no. 2, pp. 38–47, 1996. https://doi.org/10.1109/2.485845

[5] World Wide Web Consortium (W3C), "WebRTC: Real-Time Communication in Browsers," W3C Recommendation, March 13, 2025. https://www.w3.org/TR/webrtc/

[6] J. Marron, Implementing the Health Insurance Portability and Accountability Act (HIPAA) Security Rule: A Cybersecurity Resource Guide, NIST Special Publication 800-66 Revision 2, 2024. https://doi.org/10.6028/NIST.SP.800-66r2