Software Requirements Document

CareConnect

University of Maryland Global Campus

SWEN 670 – Software Engineering Capstone

Dr. Mir Assadullah

Spring 2026

Contributors: Parthav Dani, Pemon Kouadio, Mawuko Kpelevi, Jordene Downer, Melissa Burgos

Revision History

Team Name	Date	Reason for Changes	Version
CareConnect Team	01/24/2026	Initial document Submission	1.0

Software Requirements Specification (SRS)

CareConnect – Assigned Enhancements

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the functional and non-functional requirements for the Communication module of the CareConnect mobile application. This module is an enhancement to the existing CareConnect platform and is responsible for enabling secure, real-time communication between patients and caregivers.

The Communication module supports person-to-person chat messaging as well as audio and video calling capabilities. These features are intended to improve coordination, responsiveness, and continuity of care by allowing authorized users to communicate directly within the CareConnect system.

This SRS serves as a reference for system designers, developers, testers, and instructors by clearly documenting the expected behavior, constraints, and boundaries of the Communication module. It is not intended to prescribe specific user interface designs or final technology implementations, but rather to define **what the system must do** and the conditions under which it must operate.

1.2 Scope

The scope of this document is limited to communication-related functionality within the CareConnect application. Specifically, it addresses:

- Real-time person-to-person chat messaging between patients and caregivers
- Support for multiple message types, including text, image, audio, and video
- Audio and video calling between patients and caregivers
- Enforcement of role-based communication permissions
- Integration of communication services within an AWS-hosted environment

This SRS does **not** cover:

- User experience (UX) or visual interface design decisions
- AI-powered conversational features
- Clinical decision-making workflows
- Emergency response logic beyond call initiation and termination
- Payment processing and subscription management, including integration with Apple Pay, Google Pay, or third-party payment providers, are outside the scope of this SRS

All requirements defined in this document assume the presence of existing CareConnect services for authentication, user management, and patient–caregiver relationship management.

1.3 Definitions, Acronyms, and Abbreviations

The definitions, acronyms, and abbreviations used in this document are specified in Table 1 & 2.

Table 1 Document Conventions

ADL	Activities of Daily Living
AI	Artificial Intelligence
API	Application Programming Interface
AWS (Amazon Web Services)	Cloud computing platform used to host CareConnect services.
CareConnect	A mobile application that connects patients and caregivers on a unified platform to support care coordination and communication.
Caregiver	A user role responsible for providing care and initiating communication.
DMAS	Department of Medical Assistance Services (Virginia)
EVV	Electronic Visit Verification: Federal mandate to electronically confirm, capture and verify Medicaid funded home and personal care services is given to patients
HHCS	Home Health Care Services
HIPAA	Health Insurance Portability and Accountability Act
IADL	Instrumental Activities of Daily Living
IAM	Identity and Access Management
MCO	Managed Care Organization
Patient	A user role receiving care through the CareConnect platform.
PCS	Personal Care Services
PHI	Protected Health Information
RBAC	Role based access control
Signaling	The process of establishing and managing communication sessions.
SRS (Software Requirements Specification)	A formal document that defines system requirements.
WebRTC	A technology designed for real-time audio and video communication.
WebSocket	A real-time, bidirectional communication protocol suitable for persistent connections.

1.4 Document Overview

This SRS is organized to progressively describe the Communication module in increasing detail. Following the Introduction, the document outlines user roles and communication rules, followed by functional requirements for chat messaging and audio/video calling. Non-functional requirements address performance, security, and reliability considerations. The document concludes with success criteria and traceability of information to support testing and validation.

All functional requirements defined in this document are subject to the non-functional requirements specified in Section 7. These non-functional requirements apply uniformly across all communication features described herein.

2. Overall Assumptions and Constraints

This section documents the assumptions and constraints that influence the design, implementation, and operation of the CareConnect Communication module. These items define the conditions under which the system is expected to function and the limitations that must be respected throughout development.

2.1 Assumptions

The following assumptions are made for the Communication module:

- It is assumed that all users accessing communication features are authenticated through existing CareConnect identity and access management services prior to initiating any messaging or calling functionality.
- It is assumed that patient–caregiver relationships are already established and maintained by the CareConnect system. The Communication module does not create or manage these relationships but relies on them to determine authorization.
- It is assumed that user roles (e.g., patient or caregiver) are accurately defined and available to the Communication module at runtime.
- It is assumed that end-user devices (mobile phones or tablets) are equipped with the necessary hardware, such as microphones and cameras, to support audio and video communication.
- It is assumed that all communication services will be deployed and operated within an AWS cloud environment, and that supporting AWS services are available to the system.
- It is assumed that network connectivity may vary, and that users may experience temporary disconnections during communication sessions.
- Virginia Medicaid EVV requirements and submission pathways are stable for the project duration
- Caregivers and Patients will have access to a compatible smartphone or tablet

2.2 Constraints

The following constraints apply to the Communication module and must be adhered to throughout development and deployment:

- All communication functionalities must be implemented using technologies that are compatible with AWS and capable of operating entirely within AWS infrastructure.
- The system should not rely on polling-based mechanisms for real-time communication. Polling approaches previously used are considered inadequate for the required responsiveness and scalability.
- The system shall support multi-tenant architecture. All communication data, including messages and call metadata, shall be logically associated with an organization's identifier to prevent cross-organization data access.
- Communication permission must be enforced by backend business logic and not rely solely on client-side controls.
- Patients shall not be permitted to initiate communication with caregivers or doctors unless such initiation is explicitly enabled through system configuration or user profile settings.
- Real-time communication components must be designed to support persistent connections where required and avoid architectural approaches that are unsuitable for long-lived sessions.
- The Communication module must integrate with existing CareConnect services without requiring significant changes to core authentication or user management systems.

2.3 User Classes and Communication Characteristics

This section identifies the primary user classes that interact with the Communication module and summarizes their characteristics and permissions related to chat messaging and audio/video calling functionality.

Table 2- User Characteristics

User Class	Description	Chat Messaging Capabilities	Audio/Video Calling Capabilities	Key Communication Constraints
Caregiver	Authorized user responsible for providing care and coordinating with patients	Can initiate and participate in chat messaging with assigned patients	Can initiate and participate in audio and video calls, including conference calls	Communication limited to assigned patients; subject to system rules
Patient	Individual receiving care	Can participate in chat messaging;	Can participate in audio and video	Cannot initiate communication unless

User Class	Description	Chat Messaging Capabilities	Audio/Video Calling Capabilities	Key Communication Constraints
	through the CareConnect platform	may initiate messaging only if explicitly enabled	calls; call initiation may be restricted	enabled; role-based restrictions apply

User Class	Core Responsibilities	Technical Skill	Access
Caregiver	Captures visits, documents care, uses offline mode	Medium – High	Full
Patient	Receives care, log symptoms, view schedules, respond to reminders, SOS alerts and may provide electronic signatures.	Low-Medium	Limited
Family Member	Receive alerts, views logs and reports based on privileges	Low	Read- Only
Supervisor	Monitors visit status, approves visits, manages schedules, reviews reports	High	Full
System Administrator	Manages system configuration, user roles, and reference data (participants, services)	High	Full
Billing/EDI Specialist	Generates and submits claims (837P/837I) from EVV data	Medium -High	Full
EOR Approver	Authorized role to approve visits prior to billing where required	High	Limited
Compliance/Audit Officer	Accesses immutable audit logs and reports for audits.	High	Full

These user class characteristics directly inform the communication eligibility and permission rules defined in Sections 5.2 and 5.3.

2.4 Design Considerations

While this SRS does not mandate specific technologies, the following considerations influence system design:

- Lightweight, bidirectional communication mechanisms (such as WebSocket-based solutions) are preferred for chat messaging and signaling due to their efficiency and compatibility with AWS.
- Technologies optimized for real-time media exchange (such as WebRTC) may be evaluated for audio and video communication, provided they meet AWS deployment and cost constraints.
- The system should be designed to handle intermittent connectivity without data loss or system failure.

2.5 Infrastructure and Integration Assumptions

The CareConnect Communication module assumes deployment within an AWS hosted environment and relies on backend services to abstract all infrastructure, security, and AI related interactions. Backend services are expected to operate as stateless services configured through externally provided configuration parameters and managed identity mechanisms.

All AI-assisted functionality is assumed to be invoked exclusively through backend APIs, with access to AI inference services governed by managed identity and access control policies. Client applications shall not directly interact with cloud infrastructure services or AI platforms.

The frontend application consumes stable, versioned APIs exposed by backend services and is not dependent on cloud specific implementation details. This separation ensures secure operation, supports scalability, and allows the Communication module to evolve independently of underlying infrastructure implementations.

3. EVV + In-Home Residential Support

3.1 Description

The EVV (Electronic Visit Verification) features tracking caregiver visits through recording start and end times, service types, identity of caregiver/patient and location of visit. The In- Home Residential Support feature supports the documentation of in-home support activities delivered during visits, including Activities of Daily Living (ADLs) and Instrumental Activities of Daily Living (IADLs), competency scoring, behavioral tracking, medical reminders and incident reporting. This enhancement is based on an existing data dictionary and does not require backend schema changes.

3.2 EVV Functional Requirements

FR-EVV-1
The system shall support capturing EVV-related data as defined in the provided data dictionary.

FR-EVV-2
The system shall allow the EVV user interface to be implemented independently of backend changes.

3.2.1 Client & Participant Management

FR-EVV-3
The system shall allow authorized users (e.g., Administrators) to perform CRUD operations on participant records

FR-EVV-4
A participant record shall include Full Name, Medicaid Number, Date of Birth, Address, Emergency Contact, and a list of Authorized Services (PCS/HHCS).

3.2.2 ADLs/IADLs Documentation

FR-EVV-5
The system shall allow caregivers to document completed Activities of Daily Living (ADLs) and Instrumental ADLs (IADLs) during or after a visit.

FR-EVV-6
Documentation shall support free text notes and structured data entry.

3.2.3 Behavioral & Incident Tracking

FR-EVV-7
The system shall allow caregivers to record behavioral notes and incident reports.

FR-EVV-8
Incident reports shall capture time, description, actions taken, and involved individuals.

3.2.4 EVV Visit & Lifecycle

FR-EVV-9
The system shall support calendar-based scheduling of visits, capturing Patient, Caregiver, Service Type, Date, Time, Duration, and Notes.

FR-EVV-10
The system shall capture the six federally required EVV elements: Service Type, Individual

receiving care, Caregiver identity, Date of service, Time in/out (auto-captured), and Location/GPS.

FR-EVV-11
The system shall allow viewing schedules, capturing check-in/out, location, and signatures while offline, with secure queuing and auto-sync.

FR-EVV-12
The system shall support touchscreen signature capture from the patient/representative, associating it with the visit record, timestamp, and signer type.

FR-EVV-13
Caregivers shall be able to review and submit EVV records. The system shall support submission to configured Virginia aggregators/payers (e.g., Netsmart, HHAeXchange) or direct claim generation.

3.2.5 Scheduling & Oversight

FR-EVV-14
The system shall provide multiple scheduler views: Caregiver-centric and Patient-centric, with filtering by date, person, service, and status.

FR-EVV-15
Supervisors shall have a real-time dashboard showing visit status (Scheduled, Checked-in, Checked-out, Late, Missed, Pending Sync).

FR-EVV-16
The system shall generate configurable alerts for missed/late check-ins, early check-outs, and pending approvals.

FR-EVV-17
The system shall enforce an EOR approval workflow where required. Corrections shall only be made by authorized roles, requiring a reason code, comments, and shall retain original data, creating a full audit trail.

3.3 In-Home Residential Support Functional Requirements

3.3.1 Client Profile and Support Configuration

FR-IHRS-1

The system shall allow authorized users to create and maintain in-home residential support profiles for clients.

FR-IHRS-2

A client support profile shall include demographic information, support needs, risk indicators, and authorized in-home services.

FR-IHRS-3

The system shall allow administrators to configure which ADLs and IADLs apply to each client.

3.3.2 ADLs and IADLs Documentation

FR-IHRS-4

The system shall allow caregivers to document completed Activities of Daily Living (ADLs) during in-home residential support.

FR-IHRS-5

The system shall allow caregivers to document completed Instrumental Activities of Daily Living (IADLs).

FR-IHRS-6

Each ADL or IADL entry shall capture the activity type, date, time, caregiver, and client.

FR-IHRS- 7

The system shall support both structured entries (e.g., activity type, level of assistance) and option free-text notes.

3.3.3 Competency and Independence Tracking

FR-IHRS-10

The system shall allow caregivers to record behavioral observations related to in-home residential support.

FR-IHRS-11

The system shall allow caregivers to create incident reports during or after a visit.

FR-IHRS-12

An incident report shall capture the date, time, description, actions taken, and individuals involved.

FR-IHRS-13

Incident reports shall be stored as immutable records and shall not overwrite previous submissions.

3.3.4 Audit and Compliance

FR-IHRS-14

The system shall maintain an audit trail for all in-home residential support documentation

FR-IHRS-15

The system shall restrict modification of in-home residential support records to authorized roles.

3.3.5 Relationship to EVV

FR-IHRS-16

In-home residential support documentation shall be associate with an EVV-verified visit when applicable

3.4 Stimulus/Response Sequences

Stimulus: Caregiver checks in to a scheduled visit

Response: System records check-in time, captures location data, and marks the visit as in Progress

Stimulus: Caregiver documents completed ADLs or IADLs during a visit

Response: System validates required fields, associates documentation with the active visit, and stores the data securely

Stimulus: Caregiver records behavioral notes or an incident report

Response: System captures the entry with timestamp, visit reference, and user context, and stores the record

Stimulus: Unauthorized user attempts to modify a submitted EVV or in-home support record

Response: System blocks the modification and logs the attempt for audit purpose

3.5 Constraints

- Backend data structures are assumed to already exist.
- This enhancement focuses primarily on front-end implementation.
- All EVV and in-home residential support data must comply with applicable federal and state EVV requirements

4. AI Services and AWS Bedrock Integration

4.1 Overview

The CareConnect system includes AI-assisted services intended to support documentation quality, operational insight, and administrative efficiency. These services are considered an enhancement to the existing platform and are not responsible for clinical decision making or medical guidance.

This section defines the requirements and constraints associated with evaluating and integrating AWS Bedrock as the managed AI platform for CareConnect. The purpose of this enhancement is to improve data governance, security, and cost transparency by mitigating AI workloads from external providers into an AWS-hosted environment.

4.2 Scope of AI Functionality

AI functionality within CareConnect is limited to non-clinical, assistive use cases, including:

- Summarization of caregiver entered documentation
- Identification of patterns and trends in historical, non-diagnostic data
- Administrative and supervisory support functions

AI generated outputs are advisory in nature and must be reviewed by authorized users prior to use.

This SRS does not define AI functionality related to:

- Medical diagnosis or treatment recommendations
- Clinical decision support
- Autonomous system actions without human confirmation

4.3 Functional Requirements

FR-AI-1
The system shall support integration with AWS Bedrock for AI-related services.

FR-AI-2
The system shall evaluate whether to use:

- A single foundational model (e.g., Qwen-based model), or
- Individual AWS AI services (e.g., transcription, document extraction).

FR-AI-3
The system shall analyze and document the cost implications of using multiple AI models versus consolidated services.

FR-AI-4

The system shall allow AI services to be accessed only through backend APIs and not directly from client applications.

FR-AI-5

The system shall support the use of one or more AWS Bedrock supported foundation models for text-based AI functionality.

FR-AI-6

The system shall allow AI-generated content to be reviewed and confirmed by authorized users prior to persistence or submission.

FR-AI-7

The system shall ensure AI-generated outputs are clearly identifiable as system-generated content.

FR-AI-8

The system shall provide an AI assisted conversational interface that allows authorized users to submit questions related to system data, documentation, or operational guidance and receive relevant responses.

FR-AI-9

The system shall support submission of video files for automated analysis and generation of metadata or summary insights based on predefined analysis criteria.

FR-AI-10

The system shall not perform real-time monitoring or decision-making.

4.4 Model Selection and Cost Considerations

FR-AI-11

The system shall support evaluation of a single foundation model versus multiple specialized AI services based on cost, performance, and maintainability.

FR-AI-12

The system shall document cost implications associated with AI model usage, including request volume and data processing considerations.

4.5 Security and Compliance Constraints

FR-AI-13

The system shall limit AI input data to the minimum necessary to support the intended function.

FR-AI-14

The system shall ensure AI requests and responses are transmitted over an encrypted channel.

FR-AI-15

The system shall not use customer data to train or fine-tune AI models.

FR-AI-16

The system shall maintain audit logs of AI service usage, including request metadata and user context.

FR-AI-17

The system shall log AI interactions for auditability.

FR-AI-18

The system shall not provide medical or clinical advice.

FR-AI-19

The system shall give responses that are read only and informational.

FR-AI-20

The system shall support automated extraction of structured data from uploaded invoices, including vendor information, dates, line items, and totals, and store the extracted data in a machine-readable format.

FR-AI-21

The system shall make sure the extracted data is reviewable prior to submission.

FR-AI-22

The system shall retain original documents for audit purposes.

FR-AI-23

Video processing shall comply with privacy and data retention policies.

4.6 Stimulus/Response Sequences

Stimulus: User submits text for AI processing
Response: System forwards request to AWS Bedrock

Stimulus: AI generates a response
Response: System returns AI output to user

Stimulus: AI service is unavailable
Response: System returns fallback message

Stimulus: User cancels AI request
Response: System terminates processing

4.7 Constraints and Limitations

- AI services shall operate entirely within an AWS-hosted environment.
- AI functionality shall not initiate actions within EVV, billing, or clinical workflows without explicit user confirmation.
- AI services are not required to operate during offline application usage.

5. Person-to-Person Chat Messaging

The chat messaging functionality described in this section shall comply with the non-functional requirements defined in Section 7, including performance, reliability, scalability, and security constraints.

5.1 Feature Description and Priority

Priority: High

The Person-to-Person Chat Messaging feature enables secure, real-time communication between patients and caregivers within the CareConnect application. This feature supports care coordination and non-emergency communication while enforcing role-based eligibility rules and HIPAA privacy and security requirements.

All chat messages, attachments, and metadata may contain Protected Health Information (PHI) and are therefore subject to access control, encryption, audit logging, and retention requirements.

This feature operates as an enhancement to the existing CareConnect platform and integrates with existing authentication, authorization, and user-relationship services. It replaces earlier polling-based approaches with a real-time messaging mechanism capable of supporting low-latency, bidirectional communication.

5.2 Stimulus / Response Sequences

This subsection describes representative stimulus and response sequences for the Person-to-Person Chat Messaging feature. These sequences illustrate expected system behavior under normal, restricted, and failure conditions.

Stimulus: A caregiver selects an assigned patient and initiates a chat session.
Response: The system verifies caregiver authorization and communication eligibility. If permitted, the system establishes a chat session and displays the messaging interface to both participants.

Stimulus: A patient attempts to initiate a chat session with a caregiver.
Response: The system checks system configuration and user profile settings to determine whether patient-initiated messaging is enabled. If enabled, the system establishes the chat session. If not enabled, the system blocks the request and informs the patient that chat initiation is not permitted.

Stimulus: A user sends a text message within an active chat session.
Response: The system validates the message, associates required metadata, delivers the message to the recipient in near real time, and persists in the message for later retrieval.

Stimulus: A user sends an image, audio, or video message within an active chat session.

Response: The system validates the message type, associates required metadata, transmits the message content securely, and persists the message for authorized access.

Stimulus: A recipient is offline when a message is sent.
Response: The system persists in the message and delivers it when the recipient reconnects to the system.

Stimulus: A user attempts to initiate a chat session that violates configured communication rules.
Response: The system denies the chat initiation request, records the attempt for auditing purposes, and provides feedback to the user.

Stimulus: A temporary network interruption occurs during an active chat session.
Response: The system detects interruption and attempts to recover the messaging session when connectivity is restored, without loss of persisted messages.

Stimulus: A user attempts to access chat messages for which they are not authorized.
Response: The system denies access to the requested messages and enforces access control policies to prevent unauthorized viewing.

Stimulus: A user requests to view historical chat messages.
Response: The system retrieves and displays previously persisted messages associated with authorized conversations.

5.3 Use Cases Description

Table 3-Use case Specification table

Requirement ID	Requirement (Summary)	Use Case ID	Test Case ID	Verification Method
FR-CHAT-R1	Caregiver can initiate chat with assigned patient	UC-CHAT-01	TC-CHAT-01	Functional test
FR-CHAT-R2	Patient cannot initiate unless enabled	UC-CHAT-01	TC-CHAT-02	Functional test
FR-CHAT-R3	Eligibility verified before session establishment	UC-CHAT-01	TC-CHAT-03	Functional test
FR-CHAT-R4	Violations blocked and logged for audit	UC-CHAT-01	TC-CHAT-04	Functional + log review

Requirement ID	Requirement (Summary)	Use Case ID	Test Case ID	Verification Method
FR-CHAT-1	Real-time bidirectional messaging	UC-CHAT-01	TC-CHAT-05	Functional + performance observation
FR-CHAT-2	Supports text/image/audio/video message types	UC-CHAT-01	TC-CHAT-06	Functional test
FR-CHAT-3	Message metadata recorded (sender/recipient/time/type)	UC-CHAT-01	TC-CHAT-07	Data validation
FR-CHAT-4	Near real-time delivery when both connected	UC-CHAT-01	TC-CHAT-08	Functional + timing observation
FR-CHAT-5	Persist messages for offline delivery	UC-CHAT-01	TC-CHAT-09	Functional test
FR-CHAT-6	Retrieve historical messages for authorized conversations	UC-CHAT-01	TC-CHAT-10	Functional test
FR-CHAT-7	Uses AWS-compatible real-time mechanism	UC-CHAT-01	TC-CHAT-11	Inspection + deployment verification
FR-CHAT-8	Does not rely on polling-based delivery	UC-CHAT-01	TC-CHAT-12	Inspection (architecture/config)
FR-CHAT-9	Maintains persistent sessions as needed	UC-CHAT-01	TC-CHAT-13	Functional + inspection
FR-CHAT-10	Detect interruption and recover session	UC-CHAT-01	TC-CHAT-14	Functional test
FR-CHAT-11	Prevent message loss during transient failures	UC-CHAT-01	TC-CHAT-15	Functional test
FR-CHAT-12	Messages transmitted over encrypted channels	UC-CHAT-01	TC-CHAT-16	Inspection + security verification
FR-CHAT-13	Access restricted to authorized participants	UC-CHAT-01	TC-CHAT-17	Security/authorization test

5.4 Communication Eligibility and Rules

This subsection defines the rules governing who may initiate and participate in chat messaging.

FR-CHAT-R1

The system shall allow caregivers to initiate chat communication with assigned patients.

FR-CHAT-R2

The system shall restrict patients from initiating chat communication unless a caregiver has explicitly initiated the conversation.

FR-CHAT-R3

The system shall verify user role, assignment relationship, and communication eligibility on the backend before establishing a chat session.

FR-CHAT-R4

The system shall allow caregivers to mute or unmute chat communication on a per-patient basis without deleting message history.

FR-CHAT-R5

The system shall allow caregivers to communicate with other caregivers without restriction.

FR-CHAT-R6

The system shall restrict access to chat messages and attachments to authorized participants only, in accordance with HIPAA minimum-necessary access principles.

5.5 Functional Requirements

This subsection defines the functional capabilities of the chat messaging feature.

FR-CHAT-1

The system shall support real-time, bidirectional messaging between eligible users.

FR-CHAT-2

The system shall support multiple message types, including:

- Text messages
- Image attachments
- Audio message attachments (recorded clips)
- Video message attachments (recorded clips)

FR-CHAT-3

The system shall associate each message with relevant metadata, including:

- Sender identifier
- Recipient identifier
- Timestamp
- Message type

FR-CHAT-4

The system shall deliver messages in near real time when both sender and recipient are connected.

FR-CHAT-5

The system shall persist in messages to enable delivery when recipients are offline.

FR-CHAT-6

The system shall allow users to retrieve historical chat messages associated with authorized conversations.

5.6 Real-Time Communication Mechanism

This subsection defines constraints related to the communication mechanism used to support chat messaging.

FR-CHAT-7

The system shall utilize an AWS-compatible real-time communication mechanism to support chat messaging.

FR-CHAT-8

The system shall not rely on polling-based techniques for message delivery.

FR-CHAT-9

The system shall maintain persistent communication sessions as required to support real-time message exchange.

Note: Selection of a specific protocol (e.g., WebSocket-based messaging) is an implementation decision addressed during system design.

5.7 Reliability and Error Handling

This subsection defines system behavior under adverse network conditions.

FR-CHAT-10

The system shall detect temporary network interruptions and attempt to recover the messaging session when connectivity is restored.

FR-CHAT-11

The system shall prevent message loss during transient connection failures.

FR-CHAT- 12

The system shall log blocked or unauthorized chat attempts in backend audit logs.

FR-CHAT-13

The system shall retain chat messages and attachments in accordance with HIPAA retention requirements and organizational policy.

5.8 Security Requirements

This subsection defines security requirements specific to chat messaging.

FR-CHAT-14

The system shall ensure that chat messages and attachment are transmitted securely over encrypted channels.

FR-CHAT-15

The system shall restrict access to chat messages to authorized participants only.

5.9 Constraints and Limitations

The following constraints apply to the Person-to-Person Chat Messaging feature:

- Chat messaging is intended for non-emergency communication.
- The system does not guarantee message delivery during prolonged outages.
- The Communication module does not determine clinical appropriateness of messages.

5.10 Sequence Diagrams

Figure 1- Chat initiation and eligibility verification diagram

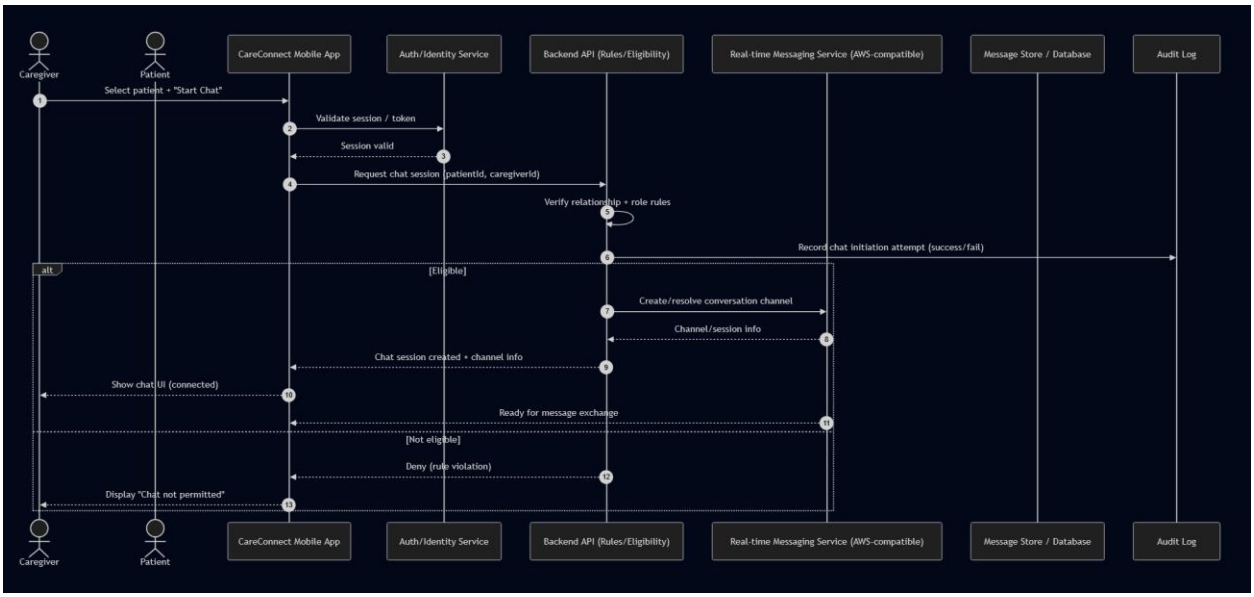
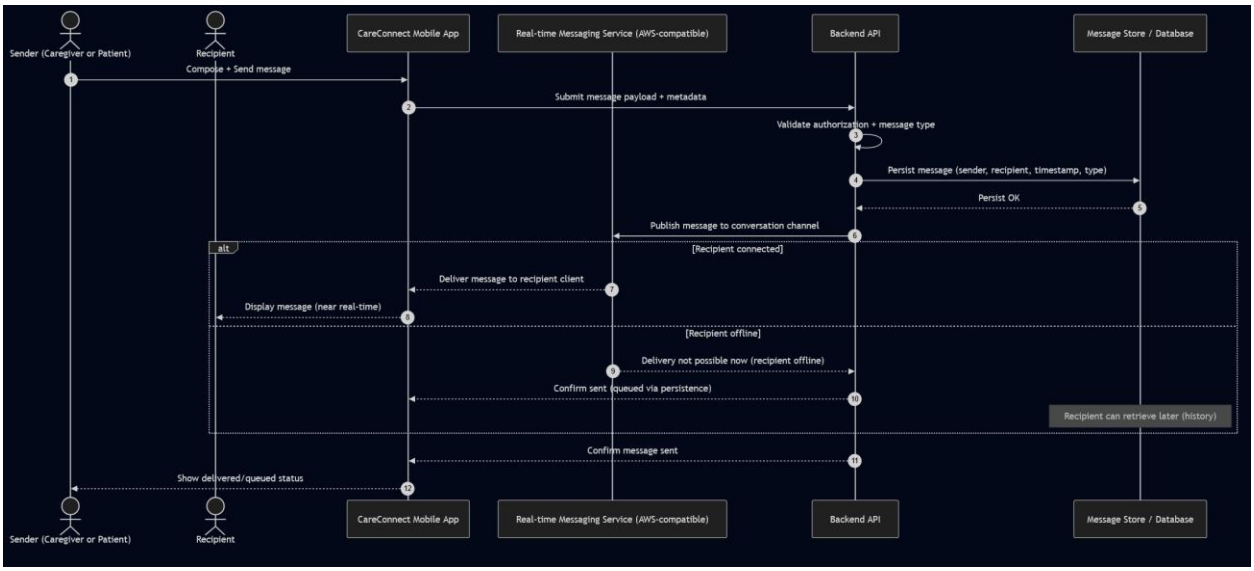


Figure 2-Send Message to connected client diagram



6. Patient/Caregiver Audio and Video Calling

The audio and video calling functionality described in this section is subject to the non-functional requirements defined in Section 7. These requirements ensure acceptable performance, reliability, scalability, and security during real-time communication sessions.

6.1 Feature Description and Priority

Priority: High

The Patient/Caregiver Audio and Video Calling feature enables real-time voice and video communication between authorized users. Unlike chat messaging, patients are permitted to initiate audio and video calls to caregivers, supporting emergency and telemedicine scenarios.

This feature extends the CareConnect Communication module by introducing live media exchange and session-based communication, while enforcing the same role-based authorization rules that govern chat messaging. The system is designed to operate within an AWS-hosted environment and to support scalable, real-time communication without relying on polling-based or stateless mechanisms.

Audio/video sessions and associated metadata may constitute PHI and must comply with HIPAA privacy, security, and retention requirements.

6.2 Stimulus / Response Sequences

This subsection describes representative stimulus and response sequences for audio and video calling.

Stimulus: A caregiver initiates an audio or video call with an assigned patient.
Response: The system verifies user roles and communication permissions. If authorized, the system initiates call signaling and notifies the recipient of the incoming call request.

Stimulus: A patient attempts to initiate an audio or video call.
Response: The system checks whether patient-initiated calling is enabled through configuration or profile settings. If enabled, the system proceeds with call setup; otherwise, it denies the request.

Stimulus: The recipient receives an incoming call request and accepts the call.
Response: The system establishes a call session prior to media exchange and begins real-time audio and/or video transmission between authorized participants.

Stimulus: The recipient declines the call or does not respond within a reasonable time window.
Response: The system terminates the call attempt and releases associated session resources.

Stimulus: A participant terminates an active call.

Response: The system ends the session, releases session resources, and records call session metadata (initiator, participants, start time, end time).

Stimulus: A network interruption occurs during an active call.
Response: The system detects the disruption and attempts to recover the call when feasible. If recovery is not possible, the system terminates the session gracefully.

6.3 Use Case Description Table

Table 4-Use case diagram table

Use Case Field	Definition
Use Case ID	UC-CALL-01
Use Case Name	Initiate and Participate in Audio/Video Call
Primary Actor	Caregiver
Secondary Actor(s)	Patient
Description	Enables authorized caregivers and patients to initiate and participate in real-time audio and video calls within CareConnect, subject to role-based permissions and eligibility rules.
Preconditions	User is authenticated; patient-caregiver relationship exists; communication permissions allow call initiation and participation.
Postconditions	Call session is terminated and resources are released; session metadata (initiator, participants, start/end time) is recorded.
Trigger	Caregiver initiates an audio or video call from the CareConnect communication interface.
Main Success Scenario	1. Caregiver initiates call. 2. System verifies roles/permissions. 3. Signaling is established. 4. Recipient is notified. 5. Recipient accepts. 6. System establishes session and begins media exchange. 7. Participants end call. 8. System releases resources and records metadata.

Use Case Field	Definition
Alternate Flows	A1: Patient initiation disabled → system denies initiation. A2: Recipient declines/no answer → system terminates attempt and releases resources. A3: Conference call invoked → system supports >2 participants when enabled.
Exception Flows	E1: Network interruption → system attempts recovery or terminates gracefully. E2: Unauthorized attempt → system blocks and logs event.
Related Requirements	FR-CALL-R1–FR-CALL-R4, FR-CALL-1–FR-CALL-6, FR-CALL-S1–FR-CALL-S4, FR-CALL-SIG1–FR-CALL-SIG4, FR-CALL-7–FR-CALL-10

6.4 Communication Eligibility and Call Initiation Rules

Audio and video calling capabilities are subject to role-based permissions to ensure appropriate use and controlled access.

FR-CALL-R1

The system shall allow caregivers to initiate audio and video calls with assigned patients.

FR-CALL-R2

The system shall allow patients to initiate audio and video calls with assigned caregivers.

FR-CALL-R3

The system shall verify user roles and communication permissions before allowing call initiation.

FR-CALL-R4

The system shall prevent unauthorized call attempts and log such events for auditing and monitoring purposes.

6.5 Functional Requirements

The system shall provide the following audio and video calling functionality:

FR-CALL-1

The system shall support real-time audio communication between authorized patients and caregivers.

FR-CALL-2

The system shall support real-time video communication between authorized patients and caregivers.

FR-CALL-3

The system shall notify recipients of incoming call requests in a timely manner.

FR-CALL-4

The system shall allow recipients to accept or decline incoming calls.

FR-CALL-5

The system shall allow any participant to terminate an active call.

FR-CALL-6

The system shall support multi-party (conference) audio and video calls involving more than two participants.

6.6 Call Session Management

Audio and video calls are managed as discrete communication sessions.

FR-CALL-S1

The system shall establish a call session prior to the exchange of audio or video data.

FR-CALL-S2

The system shall maintain session state for the duration of the call.

FR-CALL-S3

The system shall properly release session resources when a call is completed or terminated.

FR-CALL-S4
The system shall record call session metadata, including:

- Call initiator
- Participants
- Call start time
- Call end time

6.7 Signaling and Infrastructure Requirements

To support reliable call setup and coordination, the system shall employ a dedicated signaling mechanism.

FR-CALL-SIG1

The system shall use a signaling service to coordinate call initiation, acceptance, and termination.

FR-CALL-SIG2

The signaling mechanism shall support real-time message exchange suitable for call setup and control.

FR-CALL-SIG3

The system shall not rely on AWS Lambda for persistent or long-lived signaling sessions.

FR-CALL-SIG4

The signaling infrastructure shall be scalable to support multiple concurrent calls and conference sessions.

(Note: The selection of specific AWS services for signaling and media transport is an implementation decision informed by feasibility, performance, and cost considerations.)

6.8 Reliability and Fault Handling

The audio and video calling feature shall be resilient to common runtime issues.

FR-CALL-7

The system shall detect call disruptions caused by network interruptions.

FR-CALL-8

The system shall attempt to recover active calls when feasible or terminate sessions gracefully when recovery is not possible.

6.9 Security Requirements

Audio and video communication must meet CareConnect security expectations.

FR-CALL-9

The system shall encrypt audio and video data during transmission.

FR-CALL-10

The system shall restrict access to call sessions to authorized participants only.

FR-CALL-11

The system shall allow caregivers to optionally enable or disable call recording.

FR-CALL-12

Recorded calls shall be stored, protected, and retained in compliance with HIPAA requirements.

6.10 Constraints and Limitations

- Audio and video calling functionality depends on user device capabilities and network quality.
- The system does not guarantee uninterrupted calls during extended connectivity loss.
- Audio and video calling features are not intended to replace emergency services.

6.11 Sequence Diagram

Figure 3- Call setup + signaling + accept/decline

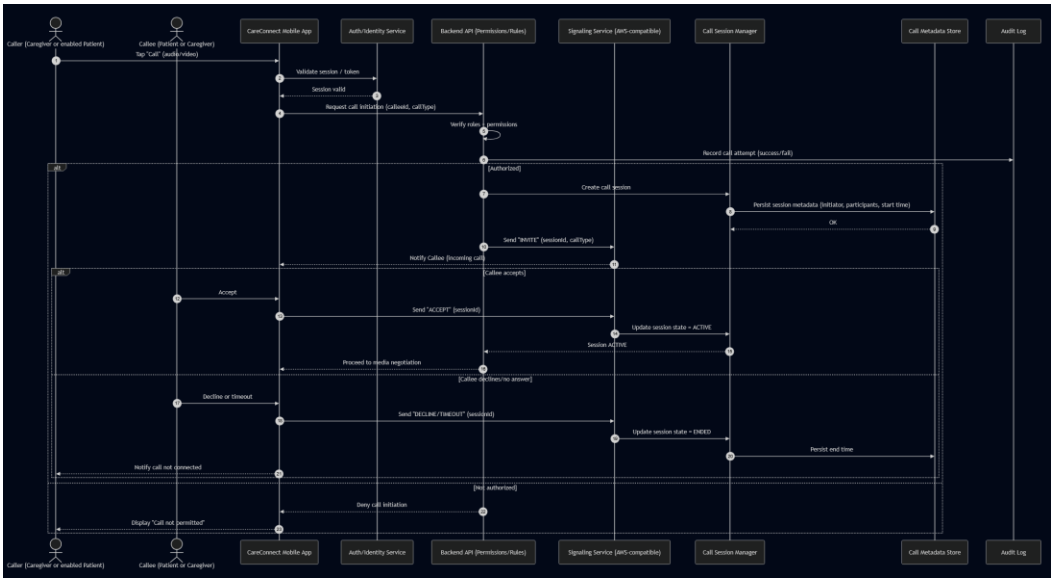


Figure 4 -Active call + terminate + fault handling diagram

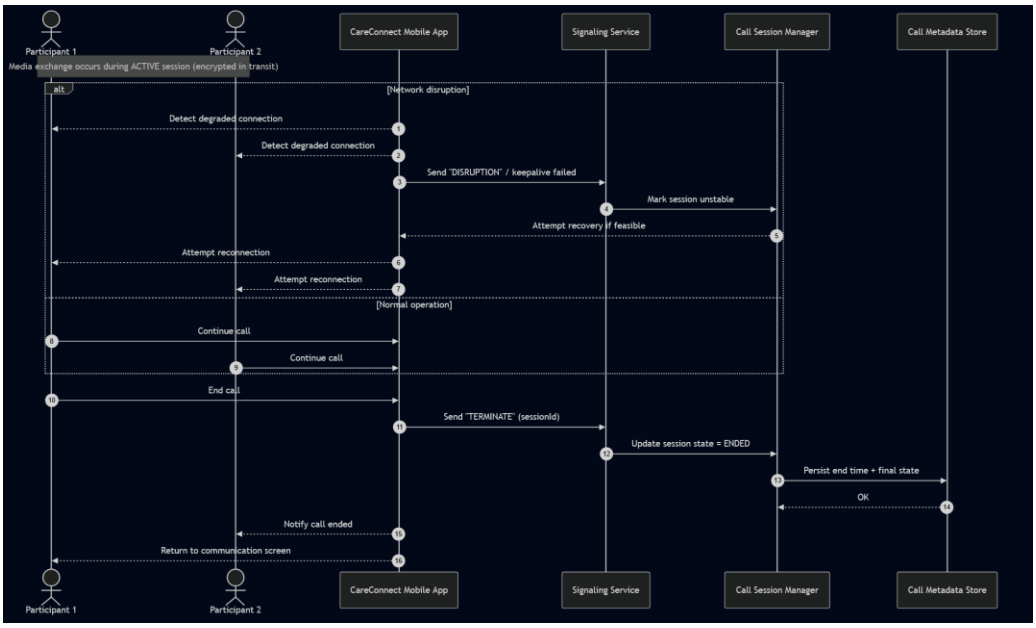
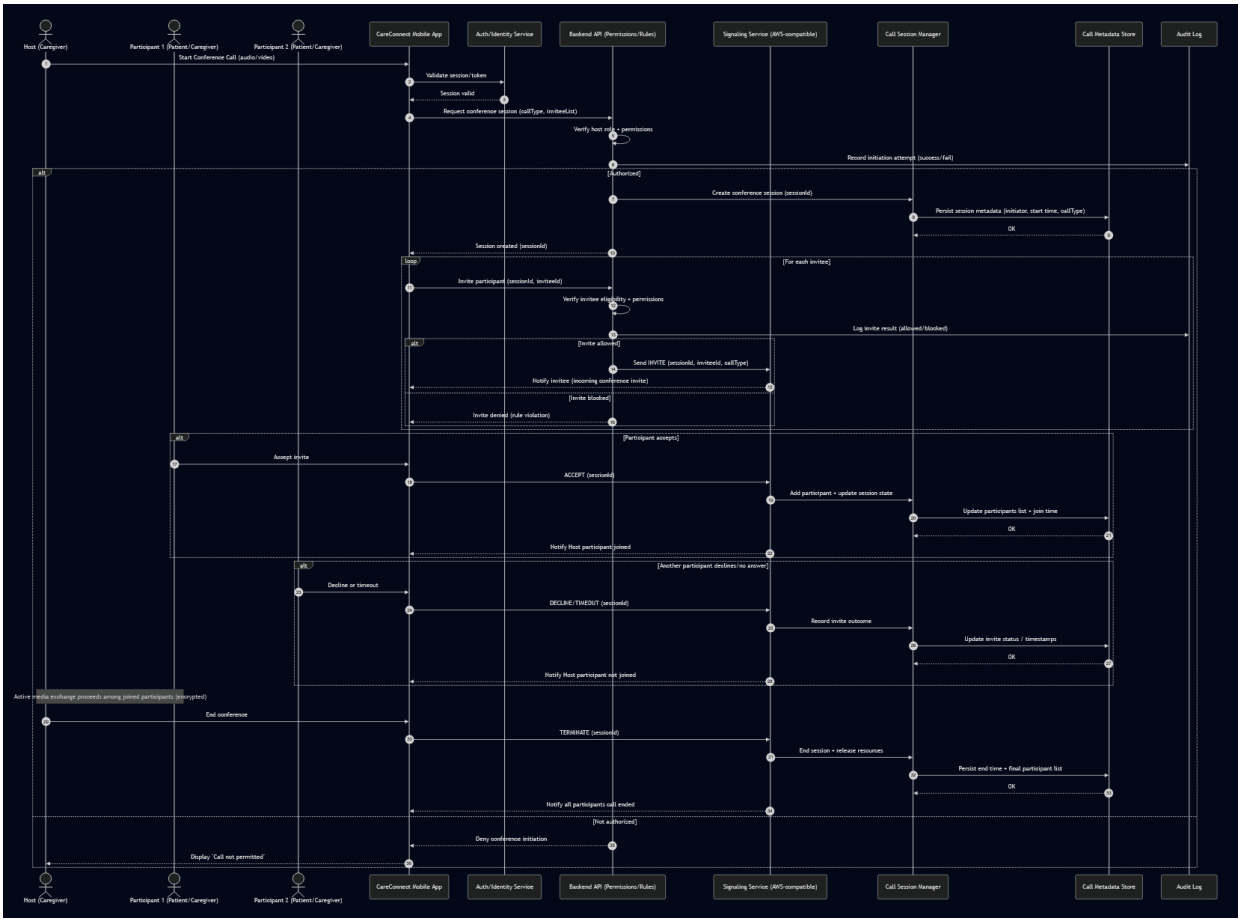


Figure 5- Conference Call (Host Invites Multiple Participants) diagram



7. Non-Functional Requirements (Communication Module)

This section defines the non-functional requirements that apply to all communication features within the CareConnect application, including person-to-person chat messaging and patient/caregiver audio and video calling. These requirements describe **how the system should perform** rather than **what the system should do**.

7.1 Performance

The Communication module must support timely and responsive interaction to ensure effective coordination between patients and caregivers.

NFR-PERF-1
The system shall deliver chat messages with minimal latency under normal network conditions.

NFR-PERF-2
Audio and video calls shall maintain acceptable quality suitable for real-time communication.

NFR-PERF-3
The system shall support multiple concurrent chat and call sessions without significant performance degradation.

7.2 Reliability and Availability

The Communication module must operate reliably in environments with varying network quality.

NFR-REL-1
The system shall handle temporary network interruptions without causing application failure.

NFR-REL-2
The system shall attempt to recover interrupted communication sessions when feasible.

NFR-REL-3
The system shall fail gracefully when recovery is not possible, ensuring that system state remains consistent.

7.3 Scalability

The Communication module must be capable of supporting growth in user volume and concurrent usage.

NFR-SCAL-1
The system shall support multiple simultaneous communication sessions.

NFR-SCAL-2
The system architecture shall allow scaling of communication services without requiring significant system redesign.

NFR-SCAL-3
Textual communication data and media content shall be handled using different storage mechanisms appropriate to their size and access patterns.

7.4 Security

Security is a critical requirement due to the sensitive nature of patient and caregiver communication.

NFR-SEC-1
All communication data shall be encrypted in transit.

NFR-SEC-2
The system shall ensure that only authorized users can access chat messages and call sessions.

NFR-SEC-3
The system shall enforce role-based access controls consistently across all communication features.

7.5 Privacy

The system must protect user privacy and limit exposure of sensitive information.

NFR-PRIV-1
Communication data shall only be accessible to users directly participating in the communication session.

NFR-PRIV-2
The system shall not expose communication content to unauthorized system components.

7.6 Maintainability

The Communication module should be designed to support ongoing maintenance and enhancement.

NFR-MAINT-1
The system shall separate communication logic from user interface logic to support maintainability.

NFR-MAINT-2
The system shall allow communication technologies to be updated or replaced with minimal impact on other system components.

7.7 Portability and Deployment

The Communication module must operate within the required cloud environment.

NFR-PORT-1
All communication services shall be deployable within an AWS-hosted environment.

NFR-PORT-2
The system shall not depend on platform-specific features that prevent deployment in AWS.

8. Requirement Traceability Matrix

Table 5-Chat messaging Traceability

Requirement ID	Requirement (Summary)	Use Case ID	Test Case ID	Verification Method
FR-CHAT-R1	Caregiver can initiate chat with assigned patient	UC-CHAT-01	TC-CHAT-01	Functional test
FR-CHAT-R2	Patient cannot initiate unless caregiver explicitly initiated conversation	UC-CHAT-01	TC-CHAT-02	Functional test
FR-CHAT-R3	Backend verifies role/assignment/eligibility before chat session	UC-CHAT-01	TC-CHAT-03	Functional test
FR-CHAT-R4	Caregiver can mute/unmute per patient without deleting history	UC-CHAT-01	TC-CHAT-04	Functional test
FR-CHAT-R5	Caregiver-to-caregiver chat allowed without restriction	UC-CHAT-01	TC-CHAT-05	Functional test
FR-CHAT-R6	Access restricted to authorized participants (HIPAA minimum necessary)	UC-CHAT-01	TC-HIPAA-01	Security/authorization test
FR-CHAT-1	Real-time, bidirectional messaging between eligible users	UC-CHAT-01	TC-CHAT-06	Functional + timing observation

FR-CHAT-2	Supports text/image/audio clip/video clip message attachments	UC-CHAT-01	TC-CHAT-07	Functional test
FR-CHAT-3	Message metadata recorded (sender/recipient/timestamp/type)	UC-CHAT-01	TC-CHAT-08	Data validation
FR-CHAT-4	Near real-time delivery when both users connected	UC-CHAT-01	TC-CHAT-09	Functional + timing observation
FR-CHAT-5	Persist messages to enable delivery when recipients offline	UC-CHAT-01	TC-CHAT-10	Functional test
FR-CHAT-6	Retrieve historical chat messages for authorized conversations	UC-CHAT-01	TC-CHAT-11	Functional test
FR-CHAT-7	Uses AWS-compatible real-time mechanism	UC-CHAT-01	TC-CHAT-12	Inspection + deployment verification
FR-CHAT-8	Does not rely on polling-based delivery	UC-CHAT-01	TC-CHAT-13	Architecture/config inspection
FR-CHAT-9	Maintains persistent sessions for real-time exchange	UC-CHAT-01	TC-CHAT-14	Functional + inspection
FR-CHAT-10	Detect interruption and attempt session recovery	UC-CHAT-01	TC-CHAT-15	Fault-injection functional test
FR-CHAT-11	Prevent message loss during transient failures	UC-CHAT-01	TC-CHAT-16	Functional test
FR-CHAT-12	Log blocked/unauthorized chat attempts in backend audit logs	UC-CHAT-01	TC-HIPAA-02	Log review
FR-CHAT-13	Retain messages/attachments per HIPAA retention + org policy	UC-CHAT-01	TC-HIPAA-03	Data retention verification

FR-CHAT-14	Encrypt chat messages/attachments in transit	UC-CHAT-01	TC-HIPAA-04	Security inspection
FR-CHAT-15	Restrict access to chat messages to authorized participants only	UC-CHAT-01	TC-HIPAA-01	Security/authorization test

Table 6-Audio/Video Traceability Matrix

Requirement ID	Requirement (Summary)	Use Case ID	Test Case ID	Verification Method
FR-CALL-R1	Caregiver can initiate audio/video calls with assigned patients	UC-CALL-01	TC-CALL-01	Functional test
FR-CALL-R2	Patient can initiate audio/video calls with assigned caregivers	UC-CALL-01	TC-CALL-02	Functional test
FR-CALL-R3	User roles and permissions verified before call initiation	UC-CALL-01	TC-CALL-03	Functional test
FR-CALL-R4	Unauthorized call attempts blocked and logged	UC-CALL-01	TC-HIPAA-06	Functional + log review
FR-CALL-1	Supports real-time audio communication	UC-CALL-01	TC-CALL-04	Functional + performance observation
FR-CALL-2	Supports real-time video communication	UC-CALL-01	TC-CALL-05	Functional + performance observation
FR-CALL-3	Notify recipients of incoming call requests	UC-CALL-01	TC-CALL-06	Functional test
FR-CALL-4	Recipients can accept or decline calls	UC-CALL-01	TC-CALL-07	Functional test

FR-CALL-5	Participants can terminate active calls	UC-CALL-01	TC-CALL-08	Functional test
FR-CALL-6	Supports multi-party (conference) calls	UC-CALL-02	TC-CALL-09	Functional test
FR-CALL-S1	Call session established before media exchange	UC-CALL-01	TC-CALL-10	Functional + inspection
FR-CALL-S2	Session state maintained for call duration	UC-CALL-01	TC-CALL-11	Functional test
FR-CALL-S3	Session resources released on call completion	UC-CALL-01	TC-CALL-12	Functional + inspection
FR-CALL-S4	Call session metadata recorded (initiator, participants, times)	UC-CALL-01	TC-CALL-13	Data validation
FR-CALL-SIG1	Uses signaling service for call coordination	UC-CALL-01	TC-CALL-14	Architecture inspection
FR-CALL-SIG2	Signaling supports real-time call setup/control	UC-CALL-01	TC-CALL-15	Functional test
FR-CALL-SIG3	Does not rely on AWS Lambda for long-lived signaling	UC-CALL-01	TC-CALL-16	Architecture inspection
FR-CALL-SIG4	Signaling scales for concurrent and conference calls	UC-CALL-02	TC-CALL-17	Load / scalability analysis
FR-CALL-7	Detect call disruptions due to network interruptions	UC-CALL-01	TC-CALL-18	Fault-injection functional test
FR-CALL-8	Recover calls or terminate gracefully when recovery fails	UC-CALL-01	TC-CALL-19	Functional test

FR-CALL-9	Encrypt audio and video data during transmission	UC-CALL-01	TC-HIPAA-07	Security inspection
FR-CALL-10	Restrict call session access to authorized participants	UC-CALL-01	TC-HIPAA-01	Security/authorization test
FR-CALL-11	Caregiver can enable/disable call recording	UC-CALL-01	TC-CALL-20	Functional test
FR-CALL-12	Recorded calls stored and retained per HIPAA requirements	UC-CALL-01	TC-HIPAA-08	Data retention verification

9. External Interface Requirements

This section describes the interfaces through which users and external systems interact with the Communication module. The purpose of this section is to identify interface-level expectations without prescribing to specific implementation or visual design details.

9.1 User Interfaces

The Communication module shall provide user-facing interfaces that enable patients and caregivers to access chat messaging and audio/video calling functionality.

The user interfaces shall support the following interactions:

- Initiating and participating in real-time chat conversations
- Sending and receiving text, image, audio, and video messages
- Initiating and responding to audio and video call requests
- Receiving notifications for incoming messages and calls
- Ending active communication sessions

User interface layout, visual styling, and accessibility considerations are outside the scope of this SRS. The interfaces are expected to integrate with existing CareConnect application screens and follow platform-specific guidelines where applicable.

9.2 Software Interfaces

The Communication module shall interface with existing CareConnect backend services and AWS-hosted communication services.

These interfaces include, but are not limited to:

- **Authentication and Authorization Services**
  Used to validate user identity, roles, and communication permissions prior to initiating chat or call sessions.
- **User Profile and Relationship Services**
  Used to determine patient–caregiver relationships and enforce communication eligibility rules.
- **Real-Time Messaging Services**
  Used to support persistent, bidirectional communication for chat messaging and call signaling.
- **Media Transport Services**
  Used to support the transmission of audio and video data during live calls.

All software interfaces shall operate within an AWS-hosted environment and comply with system security and performance requirements defined in Section 7.

9.3 Hardware Interfaces

The Communication module assumes the use of standard mobile device hardware, including microphones, cameras, speakers, and network interfaces.

The system does not directly manage hardware devices but relies on the underlying mobile operating system to provide access to required hardware capabilities.

9.4 Communication Interfaces

The Communication module shall use network-based communication interfaces to support real-time messaging and calling.

These interfaces shall:

- Support encrypted data transmission
- Maintain persistent connections where required
- Operate over standard mobile and wireless networks

Specific communication protocols are considered implementation details and are not mandated by this SRS.

10. Success Criteria

This section defines the conditions under which the Communication module will be considered complete and acceptable.

10.1 Functional Success Criteria

The Communication module shall be considered functionally successful when:

- Caregivers can initiate chat conversations with assigned patients.
- Chat messages are delivered in real time when users are online.
- Chat messages are retained and delivered when recipients reconnect.
- Caregivers can successfully initiate audio and video calls.
- Patients can receive, accept, or decline incoming calls.
- Multi-party audio and video calls can be established and terminated.
- Communication rules based on user roles and configuration settings are enforced correctly.
- The Communication module shall be considered complete when it is stable, testable, and suitable for inclusion in an application intended for submission to mobile app stores.

10.2 Non-Functional Success Criteria

The Communication module shall be considered successful from a quality perspective when:

- Communication features operate with acceptable latency and responsiveness.
- Audio and video calls maintain stable quality under normal network conditions.
- Communication data is transmitted securely.
- The system remains stable under concurrent usage.

10.3 Validation and Acceptance

Successful completion of the Communication module shall be validated through:

- Execution of test cases mapped to functional requirements
- Verification of role-based communication rules
- Demonstration of real-time chat and calling functionality
- Review of compliance with non-functional requirements defined in Section 3

11. Requirements Traceability Summary

The requirements defined in this Software Requirements Specification are structured to ensure clear traceability from stakeholder expectations to system behavior and validation activities. Each functional requirement related to chat messaging and audio/video calling is uniquely identified and mapped to corresponding design components and test cases.

Functional requirements defined in Sections 3 describe the expected system behavior for real-time communication features. These requirements are validated through test cases that verify role-based communication rules, message delivery, call initiation, and session management. Non-functional requirements defined in Section 3 apply uniformly across all communication features

and provide measurable quality attributes related to performance, reliability, scalability, and security.

This traceability approach ensures that all requirements can be verified through testing and that no implemented feature exists without a documented requirement. It also supports impact analysis by allowing changes to requirements to be evaluated against associated tests and system components.

12. Design Rationale Appendix

This appendix documents key design considerations and tradeoffs related to the Communication module. The purpose of this section is to provide context for architectural decisions without redefining system requirements or constraining implementation choices beyond what is specified in the SRS.

12.1 Real-Time Communication Technology Considerations

The Communication module requires support for real-time messaging, call signaling, and live audio/video communication. Multiple technologies were considered to meet these requirements while remaining compatible with AWS deployment constraints.

WebSocket-Based Communication

WebSocket-based communication was identified as a suitable solution for real-time chat messaging and call signaling due to the following characteristics:

- Supports persistent, bidirectional communication
- Lightweight and efficient for text-based messages and signaling events
- Well-supported within AWS infrastructure
- Lower operational complexity compared to media-focused protocols

WebSocket-based solutions are well-suited for chat messaging and for coordinating call setup, acceptance, and termination events.

WebRTC-Based Communication

WebRTC was evaluated as an option for real-time audio and video communication due to its strengths in media transport:

- Optimized for low-latency audio and video streaming
- Supports peer-to-peer communication and conference calls
- Widely adopted for real-time media applications

However, WebRTC introduces additional complexity related to signaling, network traversal, and infrastructure management. These factors require careful consideration, particularly in cloud-hosted environments.

Design Decision Summary

Based on these considerations, a hybrid approach was determined to be the most appropriate:

- **WebSocket-based communication** is preferred for:
    - Chat messaging
    - Call signaling
    - Presence and session coordination
- **WebRTC or equivalent media technologies** may be evaluated for:
    - Audio and video media transport
    - Conference calling functionality

This approach balances performance, scalability, cost, and architectural simplicity while remaining fully compatible with AWS-hosted deployment requirements.

12.2 Centralized Non-Functional Requirements Rationale

Non-functional requirements are documented in a centralized section to ensure consistent application across all communication features. Centralizing these requirements avoids duplication, simplifies validation, and improves traceability between functional requirements and quality attributes such as performance, security, and reliability.

This structure allows functional requirements to focus on system behavior,behavior, while non-functional requirements define system-wide constraints and expectations.

13. Conclusion and Final Review

This Software Requirements Specification defines the scope, behavior, and quality expectations for the Communication module of the CareConnect application. The document focuses on person-to-person chat messaging and patient/caregiver audio and video calling, with explicit consideration of role-based communication rules and AWS deployment constraints.

The requirements are structured to ensure clarity, traceability, and testability. Functional requirements describe expected system behavior, non-functional requirements define system-wide quality attributes, and success criteria establish measurable acceptance conditions. Design rationale is documented separately to provide context without constraining implementation decisions.

Together, these sections provide a complete and cohesive specification that supports implementation, testing, and evaluation of the Communication module. The document is intended to serve as a reliable reference throughout development and as a basis for validation and assessment within the course project.

CareConnect Software Requirements Document

CareConnect Team B

Version 2.1

University of Maryland Global Campus

SWEN 670 - Software Engineering Capstone

Dr. Mir Assadullah

January 27, 2026

Contributors: Eduardo Estrada, Gary Jurado, Yismaw Tilaye, Joseph Wojcik, Chastity Sapp

Table of Contents

Revision History

Team Name	Date	Reason for Changes	Version
CareConnect Team B	1/24/2026	Initial document submission.	1.0
CareConnect Team B	2/7/2026	Revisions based on 1.0 feedback	2.0
CareConnect Team B	2/22/2026	Updates based on requirement updates and system architecture updates	2.1

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the functional and non-functional requirements for CareConnect, a secure, HIPAA-compliant mobile platform designed to bridge communication between patients and caregivers. This document serves as the formal foundation for the 12-week application development cycle, providing a baseline for design, engineering, and quality assurance. CareConnect is an iterative project. This SRS distinguishes between two primary categories of requirements to ensure transparency in scope and contribution:

- **Inherited Features (Legacy):** Requirements for core modules were inherited from the previous development. These are documented to provide architectural context but are not the primary focus of Team B's engineering cycle.

- **Team B Enhancements (New/Extended):** Team B has significantly extended the legacy application by introducing:

    - **BNS 5:** Robust Offline Data Persistence and Synchronization.
    - **BNS 6:** Infrastructure Migration to AWS-Native Serverless Architecture.
    - **BNS 7:** Privacy-Preserving Observability and Telemetry.

1.2 Document Conventions (Definitions, and Acronyms)

This document uses the following conventions:

- **"Shall"** indicates a mandatory requirement.
- Requirements are labeled as **REQ-SB-#** (Supplemental Team B requirements).
- Acceptance criteria are labeled as **AC-SB-#**.

The definitions, acronyms, and abbreviations used to maintain technical precision in this document are specified in **Table 1.2.1** (Project Terms) and **Table 1.2.2** (Technical Acronyms).

Table 1.2.1

Glossary Table

Term	Definition
AI (Artificial Intelligence)	The capability of a system to perform tasks that typically require human intelligence, such as recognizing speech, making decisions, or learning from data.
Android OS	An open-source mobile operating system developed by Google, used for running applications on smartphones, tablets, and other devices.

CareConnect	Application that will be created to help manage the care receiver healthcare needs
Care Giver	A person (e.g., family member, nurse, or aide) who provides assistance, monitoring, or support to a Care Receiver through the app.
Care Receiver	The individual who requires support, assistance, or monitoring, is the primary beneficiary of the app's features.
Dart Tool	The primary programming language and toolset used to develop Flutter applications.
Diarization	The process of partitioning an audio stream containing human speech into homogeneous segments according to the identity of each speaker.
Flutter	An open-source UI software development kit (SDK) created by Google, used for building natively compiled applications for mobile (Android, iOS), web, and desktop from a single codebase.
HIPAA (Health Insurance Portability and Accountability Act)	U.S. legislation that provides data privacy and security provisions for safeguarding medical information.
iOS	A mobile operating system created by Apple Inc. for iPhones and iPads, used to run applications developed specifically for Apple devices.
Machine Learning (ML)	A subset of AI that uses algorithms and statistical models to enable systems to improve performance on tasks through experience.
NLP (Natural Language Processing)	A branch of AI that enables computers to understand, interpret, and generate human language.
User Interface (UI)	The visual and interactive components of the application which allows users (care givers and receivers) to interact with the system.
User Experience (UX)	The overall experience and satisfaction a user has when interacting with the application, including ease of use, efficiency, and accessibility.

Table 1.2.2

Document Acronyms

Acronyms	Definitions
AC-###	Acceptance criterion tested for a requirement/use Case
AI	Artificial Intelligence
API	Application Programming Interface
App	Application (mobile or web-based software)

BNS	Business Need Statement
BFF	Backend for Frontend
DB	Database
EHR	Electronic Health Record
GUI	Graphical User Interface
HIPAA/GDPR	U.S. healthcare privacy law / EU privacy regulation
IoT	Internet of Things
IaC	Infrastructure as code
ML	Machine Learning
NLP	Natural Language Processing
PHI	Protected Health Information
PII	Personally Identifiable Information
QA	Quality Assurance
RBAC	Role-Based Access Control
REQ-###	Requirement ID
REST)	Representational State Transfer (API standard)
RTO/RPO	Recovery Time/Point Objective
SRS	Software Requirements Specification
TC-###	Test case that verifies an AC
UC-###	Use case
UI	User Interface
UX	User Experience
SLA/SLO	Service Level Agreement/Objectives

1.3 Intended Audience and Reading Suggestions

Intended Audience:

- The intended audience for this SRS is Dr. Assadullah – stakeholder, Clients (Roy, Ashley), Project Manager- Alireza, Team lead- Alyssa, technical lead/architect, front and back-end developers, testers, and business analysts.
- Developers/QA: Implement/validate per AC and NFR targets.
- Security/Privacy Champion: Confirm controls and auditability.

Reading Suggestions:

- **Project Management Plan (PMP):** Defines scope, labor allocation, and risk mitigation strategies.

- **Software Requirements Specification (SRS):** Detailed functional requirements for the BNS 5, 6, and 7 technical framework.
- **Technical Design Document (TDD):** Architectural blueprint including the BNS 5 Sync Data Model and BNS 7 Privacy Middleware.
- **Programmer's Guide (PROGRAMMERS_GUIDE.md):** Implementation constraints, AWS-native architecture, and development standards for future cohorts.
- **Software Test Plan & Test Report:** Execution records for unit, integration, and HIPAA-compliance validation.
- **Deployment and Operations Guide:** SOPs for AWS CloudFormation environment provisioning and secret management.
- **User Guide (USER_GUIDE.md):** Visual guides for offline queues, EVV workflows, and calendar management.
- **Caregiver Domain Context:** https://www.hopkinsmedicine.org/about/community-health/johnshopkins-bayview/services/called-to-care/what-is-a-caregiver

1.4 Product Scope

The overall vision for CareConnect is to establish a shared, collaborative ecosystem for caregivers and patients, ensuring a unified approach to health management. To ensure high-reliability delivery within the constraints of a 12-week capstone cycle and a serverless budget model.

The scope of this iteration is strictly defined by three core Business Needs: Offline Persistence (BNS 5), AWS-Native Infrastructure (BNS 6), and Privacy-Preserving Observability (BNS 7). All deliverables defined below are developed to a production-ready standard, as governed by the quality requirements in the Definition of Done (PMP Section 5.5). For the comprehensive scope of all inherited baseline features refer to the legacy SRS Section 1.4 [2, 3].

1.4.1 In-Scope

Note: Below In-Scope and Out-of-Scope are from Project Plan. It has detail explanation of the bullet point

1.4.1.1 Core User & Profile Management [2, 3]

- The system shall allow caregivers to register, authenticate, and manage patient profiles.
- The system shall support Role-Based Access Control (RBAC) to ensure data security between caregiver and patient views.

1.4.1.2 Essential Care Modules [2, 3]

- **Tasking & Medication:** Caregivers shall assign daily tasks (medication/meal reminders); patients shall mark tasks as complete.
- **Health Data Logging:** Manual recording of vital signs and health data with report generation.
- **Calendar:** A shared calendar for manual entry of events, tasks, and reminders.

1.4.1.3 Offline-First Data Integrity (BNS 5)

- The system shall implement a data persistence abstraction layer to enable offline functionality for a limited subset of high-value modules during this term.
- The Due to scope adjustments and the addition of the "Start Normalization of Codebase" requirement, full offline synchronization is implemented for two representative modules:
  - Mood & Symptom Tracking
  - Calendar (Manual entries)
- The Modules outside the approved offline scope (including Medication Logging and other legacy modules) may leverage the shared offline framework for local capture only, but are not fully synchronized within this development cycle.
- The system shall implement a sync-on-reconnect mechanism to ensure data consistency with the cloud-native backend.
- The system shall provide a Manual Offline Toggle within the application settings, allowing the user to explicitly enable or disable "Offline Mode" to manage data usage or sync timing.

1.4.1.4 Infrastructure Automation (BNS 6)

- The system shall transition all infrastructure provisioning from legacy configurations to a standardized AWS CloudFormation framework.
- The system shall use automated templates to standardize workflows, ensuring reproducible and version-controlled deployments.
- The migration shall preserve all existing networking, compute, storage, and security configurations established in legacy documentation [2, 3].

1.4.1.5 Observability & Privacy-Preserving Analytics (BNS 7)

- **Operational Monitoring:** The system shall implement distributed tracing and centralized logging to enable rapid troubleshooting and performance monitoring.
- **Anonymous Feature Analytics:** The system shall implement a mechanical capture mechanism to track feature usage patterns.
- **Privacy Constraint:** Analytics collection shall be strictly anonymous; a middleware layer shall be implemented to ensure PII/PHI is not collected for telemetry.

1.4.1.6 Engineering Continuity

- The system shall include documented rationales for the architectural designs to support future scalability and knowledge transfer.

1.4.1.7 Start Normalization of Codebase (New Graded Requirement)

- The system shall begin normalization of the codebase to establish a professional foundation for the application to function reliably. This includes standardizing API pathing, reducing duplicate controllers/DTOs, and consolidating project structure. This effort is scope-limited and distributed across teams, Team B sets core standards and unblocks shared frameworks.

1.4.2 Out-of-Scope

The following items are explicitly excluded from the current CareConnect development cycle to ensure project feasibility, minimize third-party integration risks, and focus labor on team B's core technical requirements (BNS 5, 6, and 7).

1.4.2.1 Clinical Diagnosis & Decision Support [2, 3]

- The system shall not offer medical diagnoses, prescribe treatments, or provide clinical decision support. All AI responses remain informational.

1.4.2.2 Accessibility & Language Enhancements [2, 3]

- The application shall not include specialized accessibility features such as voice control.
- The system shall not support American Sign Language (ASL) translation.

1.4.2.3 Home Automation & Environmental Control [2, 3]

- The system shall not link to or monitor smart home devices, cameras, or home sensors (e.g., motion detectors, smart lighting, or climate control).

1.4.2.4 Custom Hardware & Wearables Integration [2, 3]

- The project shall not involve the development of new wearables or proprietary medical hardware.
- Direct integration with Fitbit, Apple HealthKit, and Google Health Connect is excluded.

1.4.2.5 Financial, Billing, & Mail Management [2, 3]

- The system shall not include tiered subscription billing or external bill assistant features (e.g., invoice scanning).
- Integration with USPS Informed Delivery or any mail-digitization services is excluded.

1.4.2.6 Full Offline Application Parity (BNS 5)

- The application shall not provide full offline functionality for all features. Offline support is strictly limited to the two modules defined in Section 1.4.1.3.

1.4.2.7 Non-AWS or Multi-Cloud Environments

- Infrastructure-as-Code (IaC) is restricted to AWS-native tools; support for Terraform or multi-cloud providers is excluded.

1.4.2.8 Advanced Business Intelligence

- The CareConnect application shall not include predictive analytics or user behavior modeling based on telemetry data.

1.4.2.9 AI Feature Enhancement

- The system shall not include new prompt engineering, model fine-tuning, or natural language processing (NLP) development.
- Development is strictly limited to the infrastructure migration to AWS Bedrock.

1.4.2.10 Full Offline Synchronization for All Modules (BNS 5)

- Full offline synchronization for all supported modules is deferred to future iterations.
- For this term, offline synchronization is implemented for a limited subset of modules to validate the synchronization framework while prioritizing codebase normalization and architectural stabilization.
- Remaining modules leverage the shared offline framework but do not perform end-to-end synchronization.

1.5 References

1. Amazon Web Services. (2026). *AWS Pricing Calculator*. https://calculator.aws/
2. Angeles, D., Bias, T., Gaucin, J., Chen, F., Curran, L., Grbreegziabhere, A., Harding, A. M., Malla-Paudel, A., Ramirez, M., Raphael, E., Truong, D., Vecchioni, A., & Yawn, C. (2025). *Software Requirements Document: CareConnect* [Legacy Documentation]. University of Maryland Global Campus. https://umgc-cappms.azurewebsites.net/previousprojects
3. CareConnect Developer Team. (2025). *CareConnect project documentation and programmer guides*. GitHub. https://github.com/umgc/2025_fall/tree/developer/careconnect2025/docs/project-docs
4. Gemini 3 Flash. (2026). *CareConnect System Architecture Diagram* [AI-generated image]. Google AI. https://gemini.google.com/
5. Gies College of Business. (n.d.). *Accessibility standards: WCAG 2.1*. University of Illinois. https://publish.illinois.edu/accessibility-training/accessibility-standards-wcag-2-1/
6. Mir, A. (2025). *CareConnect high level requirements* [Class Lecture/Handout]. University of Maryland Global Campus.
7. OpenAI. (2025). *ChatGPT* (May 30 version) [Large language model]. https://chat.openai.com/
8. U.S. Bureau of Labor Statistics. (2024, April). *Occupational outlook handbook: Software developers, quality assurance analysts, and testers*. https://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm
9. van der Meij, J., & van der Meij, H. (2014). A comparison of paper-based and video tutorials for software learning. *Computers and Education, 78*, 150–159. https://doi.org/10.1016/j.compedu.2014.06.003
10. World Wide Web Consortium. (2025). *Web Content Accessibility Guidelines (WCAG) 2.1*. W3C Recommendation. https://www.w3.org/TR/WCAG21/

2. Overall Description

This Supplemental SRS covers Team B enhancements to the existing CareConnect product. CareConnect is a mobile-first, cloud-native application built with Flutter (cross-platform) and an AWS-hosted backend designed to support healthcare-related workflows. Two primary user roles are supported and are impacted by Team B enhancements:

- **Caregiver (professional or family)** – uses scheduling, tasks, and care coordination features; may need to record visits (EVV) and use the calendar when connectivity is unreliable.
- **Patient (care receiver)** – logs wellness information (mood, symptoms, medications) and may use these features in environments with intermittent network access.

Team B's enhancements do not change the core product purpose or workflows. Instead, they add platform capabilities that improve reliability and maintainability:

- **Limited offline capability** for selected modules (Mood & Symptom Tracking and Manual Calendar)
- **Infrastructure standardization** by migrating Team B-owned provisioning from Terraform to **AWS CloudFormation**
- **System observability** (logs, metrics, traces) and **anonymous feature usage analytics** that explicitly avoid PII/PHI collection

Constraints (Supplemental):

- Must be delivered within the **capstone schedule** and align with existing system architecture and release planning.
- Must meet **HIPAA/GDPR minimum expectations** by avoiding inappropriate collection of sensitive data and applying secure transport/storage practices.
- Must remain cost-conscious and leverage cloud-native or managed services already approved for the CareConnect legacy application.

2.1 Product Perspective

Team B enhancements are implemented as supporting components within the existing CareConnect platform. The mobile client remains a Flutter application deployed on iOS/Android, and the backend remains AWS-hosted. Team B's work attaches to this architecture in three ways:

1. **Offline Data Storage and Sync (Mobile-Side Enhancement)**

   Team B adds an offline-capable data layer in the mobile application that allows selected modules to work when the device is offline or cannot reach backend services. This includes:
   a. Local persistence of only the essential data elements needed for offline operation
   b. A synchronization workflow that uploads locally stored changes once connectivity is restored
   c. A clear separation (abstraction layer) so future teams can extend offline support without rebuilding storage/sync logic

2. **Infrastructure as Code Standardization (Cloud-Side Enhancement)**

   Team B migrates Team B-owned infrastructure provisioning from Terraform to AWS CloudFormation. CloudFormation templates become the approved, reproducible source for Team B-managed AWS resources and deployment consistency.

3. **Observability and Anonymous Feature Analytics (Platform Enhancement)**

   Team B adds or improves system monitoring by collecting:
   a. Logs (to understand errors and system events)
   b. Metrics (to understand performance and reliability trends)
   c. Traces (to follow requests across services)
   Separately, Team B defines anonymous feature usage events to understand which features are used and how often—without capturing PII/PHI.

These enhancements are designed to integrate with existing deployment pipelines and operational practices already established for CareConnect.

2.2 Product Functions

The CareConnect application is structured around high-reliability health coordination and infrastructure stability. The following table summarizes the core product features categorized by functional domain and user access.

Table 2.2.1

Core Product Features

Category	Feature	Description	User Role(s)
Onboarding & Auth	Registration & Login	Secure access via AWS Cognito (Email, SSO). [2, 3]	Patient, Caregiver
	Account Recovery	Standard password reset and session management. [2, 3]	Patient, Caregiver
User & Role Mgmt	Patient-Caregiver Link	Connection via QR Code or unique Invite ID. [2, 3]	Patient, Caregiver
	Family Access	Read-only permissions for wellness monitoring. [2, 3]	Family Member
Scheduling & Alerts	Task Manager	Creation and tracking of med/meal schedules. [2, 3]	Caregiver
	Local Notifications	Automated push reminders for daily tasks. [2, 3]	Patient, Caregiver
Health Tracking	Data Entry Logs	Input forms for mood, symptoms, and vitals. [2, 3]	Patient
	Trend Analytics	Visual graphs representing health status over time. [2, 3]	Caregiver

Emergency Support	SOS Signaling	One-touch alert broadcasting GPS location. [2, 3]	Patient
BNS 5: Offline Sync	Local Storage	Persistence of health logs without connectivity.	Patient
	Automated Sync	Background mirroring of local data to AWS RDS.	System
BNS 6: Infrastructure	CloudFormation Stacks	Automated, AWS-native environment deployment.	System (Admin)
BNS 7: Observability	Telemetry & Logs	PII-scrubbed system health monitoring.	System (Admin)
	Anonymous Analytics	Tracking feature adoption without PII/PHI.	System (Admin)

2.3 User Classes and Characteristics

CareConnect is designed for three primary user groups, ensuring that security frameworks and accessibility correlate with clinical and practical applications.

1. **Caregivers (Primary Users):** Manage schedules, monitor health data, and coordinate emergency responses. They act as the "Administrative" layer of the patient's care circle.
2. **Patients (Care Recipients):** Utilize the mobile application for autonomous health management, marking tasks complete, and logging daily wellness metrics.
3. **Family Members (Secondary Users):** Possess "Read-Only" permissions to ensure transparency and involvement in the care process without the ability to modify medical schedules or clinical data.

2.4 Operating Environment

CareConnect is designed as a mobile-first application, with its primary deployment targeted at modern smartphones and tablets. The system must support both Android and iOS platforms using a shared codebase built with Flutter, ensuring consistency and reduced maintenance overhead. Team B enhancements operate
within the existing CareConnect environments:

- Mobile: iOS 16+/Android 10+.
- Web: Evergreen browsers (last 2 versions).
- Hosting Infrastructure: Amazon Web Services.

2.4.1 Hardware Requirements

- Mobile Devices: Minimum supported specifications include smartphones with ARM64 processors, at least 2 GB RAM, and modern camera capability for QR onboarding.
- Device Services: Offline Functionality may rely on GPS and device time services for EVV timestamps and ordering of offline events.
- Note: Wearables and smart home device integrations are out-of-scope for Team B in this iteration (see Section 1.4.2.3 and 1.4.2.4).

2.4.2 Software Requirements

- Operating Systems: iOS 15.0+ and Android 11.0+

- Framework: CareConnect is built using the Flutter SDK and Dart programming language to ensure cross-platform compatibility for mobile applications.
- Third-Party Integrations:
    - None are introduced or expanded under Team B scope for this iteration.
    - Any legacy integrations remain governed by legacy scope and are not modified by Team B (see Section 1.4.1 and legacy references [2, 3]).
- AWS Services:
    - Amazon Cognito for authentication and authorization
    - Amazon S3 for encrypted media and document storage
    - Amazon EventBridge for scheduling and event-driven workflows
    - Amazon RDS for PostgreSQL and Amazon Aurora (MySQL and PostgreSQL) for managed relational data storage
    - Observability backend selected per deployment environment, with OpenTelemetry used as the standard instrumentation layer
- Auth 2.0 for secure integration with wearable devices and camera systems
- CI/CD and Infrastructure:
- CodeMagic for mobile application continuous integration and deployment
- GitHub Actions for backend service pipelines
- AWS CloudFormation for infrastructure as code (IaC) management
- OAuth 2.0 for secure third-party and device integrations

2.4.3 Network Requirements

- A minimum 1 Mbps upload speed is recommended to support video calls, uploads, and background API calls.
- Real-time features like notifications and calls require maximum 300ms latency.
- A stable internet connection is required for syncing data, receiving push notifications, and media uploads. Intermittent offline caching is supported for schedules, notes, and basic data.

2.4.4 System Environment

- AWS serverless-first deployment with scale-to-zero principles.
- TLS 1.3 encryption, HIPAA-eligible services (Cognito, RDS, S3).
- Monitoring: OpenTelemetry-based instrumentation with environment-specific exporters for logs, metrics, and traces.
- Auto-scaling for resilience and cost efficiency.
- All commits must be automatically deployed to an isolated AWS test environment for integration and compliance testing prior to advancement.

2.4.5 Environmental Constraints

- Partial offline functionality is available for caregivers to access cached patient data, schedules, and care instructions.
- UI/UX design must account for varying screen sizes, limited bandwidth, and potentially low device performance on older phones.

- Basic accessibility features such as screen reader compatibility and voice input are considered, but full compliance with accessibility standards is deferred to future releases; full WCAG 2.1 Level AA deferred.

2.5 Design and Implementation Constraints

Team B work is constrained by:

Offline Scope Constraint

- Offline capability is **explicitly limited** to:
- Mood & Symptom tracking
- Manual Calendar

Local storage must be limited to **essential data only**, not a full backend mirror.

Technology Constraint

- Mobile offline storage must use a **Flutter-compatible** local storage approach.
- Offline storage and sync must be implemented behind an **abstraction layer** to reduce coupling.

Infrastructure Constraint

- Team B infrastructure must be defined and maintained using **AWS CloudFormation**.
- Terraform must not be used for Team B resources after migration is complete.

Security/Privacy Constraint

- Observability and analytics must not include **PII/PHI** in logs, traces, metrics, or analytics events.
- Data must be transmitted over secure channels and stored according to existing platform security rules.

Project Constraint

- Enhancements must be deliverable under the capstone schedule and fit within existing cost constraints and approved AWS services.

2.6 User Documentation

Team B enhancements rely on and extend existing documentation practices:

- The **User Guide** should describe offline expectations in plain language (for example: how the app behaves when offline, what features work offline, and what happens when connectivity returns).

- The **Programmer's Guide** should document:
  - the offline storage abstraction layer and schema scope
  - the synchronization workflow and conflict-handling approach
  - CloudFormation templates, parameters, and deployment steps
  - telemetry event definitions and privacy rules (what is allowed vs forbidden)

These documents are used for user support, onboarding future cohorts, and ensuring consistent implementation.

2.7 Assumptions

Assumptions

- Users may have intermittent connectivity; offline use is expected to occur in real-world scenarios.
- The backend APIs support synchronization patterns (for example: idempotent writes or safe retries).
- The mobile OS provides secure storage capabilities needed for local persistence.

Dependencies

- Offline syncing depends on network availability returning and secure communication channels functioning.
- Infrastructure migration depends on access to AWS accounts and permissions required to provision CloudFormation stacks.
- Observability depends on the availability of logging/metrics/tracing tools and that backend services are instrumented consistently.
- Anonymous analytics depends on agreed event definitions and enforcement that payloads do not contain PII/PHI.

2.8 Dependencies

CareConnect relies on the external technologies, frameworks, and services established in the legacy documentation [2, 3]. Supplemental dependencies for the current project phase include:

- **BNS 5 (Offline):** Dependent on mobile OS-level encryption (iOS Keychain/Android Keystore) for secure local Drift/SQLite persistence.
- **BNS 6 (Infrastructure):** Dependent on AWS IAM permissions required for **CloudFormation** stack provisioning and resource management.
- **BNS 7 (Observability):** Dependent on consistent backend instrumentation for CloudWatch/X-Ray telemetry and PHI redaction middleware

Team A Dependency: Offline implementation for EVV is coordinated with Team A. Team B provides the offline framework standards and integration expectations, while Team A implements EVV-specific offline workflows under the shared strategy.

2.9 Constraints

The project adheres to the non-negotiable limitations defined in the legacy baseline **[2, 3]**. The following technical constraints are specific to the Team B implementation:

- **Infrastructure (BNS 6):** All provisioning shall transition from legacy Terraform to AWS CloudFormation to standardize the environment.
- **Architecture:** Mandatory serverless-first and scale-to-zero principles for all new and refactored compute resources.
- **Data Integrity:** Offline functionality is constrained to the two specified modules (Mood & Symptom Tracking & Manual Calendar) and requires a manual user toggle.
- **Compliance:** While HIPAA/GDPR standards are mandatory, full accessibility conformity and wearable integration beyond Fitbit/Nest are deferred to future terms.
- **API Standardization:** All new or refactored endpoints shall follow the "/v3/api/…" path convention, and teams shall reuse existing Controllers and DTOs when available to prevent redundant code.

3. Specific Requirements

This section lists the Team B enhancement features for CareConnect and gives a brief description of them. These features are grouped into modules where similar capabilities are organized together. Team B enhancements focus on offline capability, infrastructure provisioning standards, and observability + anonymous feature analytics. Team B does not change CareConnect's clinical workflows, billing logic, or authentication.

3.1 Offline Data Storage and Synchronization

3.1.1 Description

This feature enables approved modules of the CareConnect mobile application to function when internet connectivity is unavailable. The system shall store limited approved data locally on the device and automatically synchronize that data with backend services once connectivity is restored. Offline capability is restricted to defined high-value modules to maintain security, compliance, and application performance.
As part of a scope adjustment approved by course instructors, offline synchronization during this term is limited to two representative modules. This approach validates the synchronization framework while allowing Team B to allocate effort toward codebase normalization and architectural consistency. The framework is designed to support expansion to additional modules in future iterations without rework.

3.1.2 Functional Requirements

- REQ-SB-101: The system shall store approved offline data locally on the user's device when network connectivity is unavailable. Priority: High
- REQ-SB-102: The system shall restrict offline synchronization during this term to a limited subset of approved modules (Mood & Symptom Tracking and Manual Calendar). Other approved modules may leverage the offline framework but are not required to fully synchronize within this development cycle. Priority: High
- REQ-SB-103: The system shall encrypt all locally stored offline data at rest using AES-256 or an equivalent platform-supported encryption standard. Priority: High
- REQ-SB-104: The system shall automatically detect restoration of network connectivity. Priority: High
- REQ-SB-105: The system shall synchronize locally stored data with backend systems upon restoration of connectivity. Priority: High
- REQ-SB-106: The system shall prevent duplication of records during synchronization. Priority: High
- REQ-SB-107: The system shall resolve synchronization conflicts using a timestamp-based Last-In-Wins reconciliation strategy. Priority: Medium
- REQ-SB-108: The system shall limit local storage to essential data elements required for offline-supported modules and shall not replicate the full backend dataset. Priority: High
- REQ-SB-109: The system shall log synchronization events for monitoring and troubleshooting purposes. Priority: Medium
- REQ-SB-110: The system shall notify the user if offline data synchronization fails. Priority: Medium
- REQ-SB-111: The system shall detect offline status using device connectivity checks (network reachability), not solely based on a manual UI toggle. Priority: High
- REQ-SB-112: The system shall allow users to pause or resume synchronization while online, without changing the offline detection state. Priority: Medium
- REQ-SB-113: The system shall maintain a durable synchronization queue for offline-created or offline-modified records and shall replay the queue upon reconnection until all queued items are confirmed synchronized or flagged as failed. Priority: High
- REQ-SB-114: The system shall process the sync queue in a deterministic order (for example FIFO by timestamp) and shall use idempotent submission to prevent duplicates. Priority: High

3.2 Infrastructure Migration (Terraform → AWS CloudFormation)

3.2.1 Description

This feature standardizes all Team B AWS infrastructure provisioning using AWS CloudFormation. All AWS resources within Team B scope shall be defined as Infrastructure as Code (IaC) templates stored in version control and deployed through CI/CD pipelines. Terraform shall no longer be used for Team B resources.

Observability and feature usage analytics are intentionally separated. Observability captures operational signals required for debugging and reliability (logs, metrics, traces). Anonymous feature analytics captures high-level usage signals to understand adoption. These systems use different event types, different storage, and different access controls to reduce privacy risk and prevent scope creep.

3.2.2 Functional Requirements

- REQ-SB-201: The system shall provision Team B AWS resources using AWS CloudFormation templates. Priority: High
- REQ-SB-202: The system shall store all CloudFormation templates in a version-controlled repository. Priority: High
- REQ-SB-203: The system shall deploy CloudFormation stacks through an automated CI/CD pipeline. Priority: High
- REQ-SB-204: The system shall remove Terraform configurations associated with Team B-managed AWS resources after CloudFormation parity is achieved. Priority: High
- REQ-SB-205: The system shall support automated rollback of failed CloudFormation stack deployments. Priority: High
- REQ-SB-206: The system shall ensure infrastructure deployments are reproducible across development, testing, and production environments. Priority: High
- REQ-SB-207: The system shall validate CloudFormation templates prior to deployment. Priority: Medium

3.3 Observability and Anonymous Feature Analytics

3.3.1 Description

This feature provides centralized observability capabilities to support operational monitoring, troubleshooting, and performance analysis. The system shall collect structured logs, system metrics, and distributed traces while maintaining compliance with healthcare data protection standards.

3.3.2 Functional Requirements

- REQ-SB-301: The system shall collect centralized application logs for backend services. Priority: High
- REQ-SB-302: The system shall collect system performance metrics, including CPU utilization, memory usage, request latency, and error rates. Priority: High
- REQ-SB-303: The system shall implement distributed tracing across service boundaries. Priority: High
- REQ-SB-304: The system shall retain operational logs for a minimum retention period consistent with compliance requirements. Priority: High
- REQ-SB-305: The system shall restrict access to observability data using role-based access control (RBAC). Priority: High

- REQ-SB-306: The system shall generate alerts when defined performance or error thresholds are exceeded. Priority: Medium
- REQ-SB-307: The system shall ensure observability data does not expose protected health information (PHI). Priority: High

3.4 Anonymous Feature Usage Analytics

3.4.1 Description

This feature enables the collection of anonymous feature usage analytics to support product decision-making without collecting personally identifiable information (PII) or protected health information (PHI). Analytics shall be enabled by default with a one-time notice and an explicit opt-out control available at any time in Settings. Analytics events shall be persisted in a dedicated database table (separate from clinical tables) to support review, auditing, and aggregation. Storage shall support time-bounded retention and privacy review.

3.4.2 Functional Requirements

- REQ-SB-401: The system shall collect anonymous feature usage events for approved application features, and shall present a one-time notice explaining analytics collection and user control. Priority: High
- REQ-SB-402: The system shall ensure that analytics events do not contain PII or PHI. Priority: High
- REQ-SB-403: The system shall record feature usage frequency and interaction counts. Priority: Medium
- REQ-SB-404: The system shall record feature adoption metrics, including feature activation rates. Priority: Medium
- REQ-SB-405: Authorized administrators shall be able to query aggregated analytics data through database queries or backend tooling without requiring a real-time dashboard. Priority: Medium
- REQ-SB-406: The system shall store analytics data separately from clinical data repositories. Priority: High
- REQ-SB-407: The system shall support configurable retention policies for analytics data. Priority: Medium
- REQ-SB-408: The system shall persist approved anonymous telemetry events into a dedicated database table that is logically separated from clinical/health data tables. Priority: High
- REQ-SB-409: The system shall allow users to opt out of anonymous feature analytics at any time via application settings, and when opted out, the system shall not emit analytics events. Priority: Medium

3.5 Acceptance Criteria

This section defines measurable and testable acceptance criteria for all Section 3 requirements (REQ-SB-###). Each acceptance criterion is uniquely identified and directly traceable to its corresponding requirement through the Requirements Traceability Matrix (Appendix A). All acceptance criteria are written to support objective verification through defined test cases.

Table 3.5.1
Offline Data Management Acceptance Criteria

AC ID	Title	Priority	Related Requirement	Given	When	Then
AC-SB-OFF-001	Offline Data Persistence	High	REQ-SB-101	The device has no internet connectivity	A user submits data from an approved offline module	The system shall store the data locally without error or data loss
AC-SB-OFF-002	Module Restriction Enforcement	High	REQ-SB-102	A module is not designated as offline-approved	The user attempts to store data offline	The system shall prevent full offline synchronization for unsupported modules and display an appropriate notification indicating limited offline support
AC-SB-OFF-003	Encrypted Local Storage	High	REQ-SB-103	Data is stored locally on the device	The storage mechanism is examined	The data shall be encrypted at rest using platform-supported encryption standards
AC-SB-OFF-004	Connectivity Detection	High	REQ-SB-104	The device transitions from offline to online state	Connectivity is re-established	The system shall detect the restored connection without requiring user action

AC-SB-OFF-005	Automatic Synchronization on Reconnect	High	REQ-SB-105	Locally stored offline data exists	Network connectivity is restored	The system shall automatically synchronize the data with backend services within 30 seconds
AC-SB-OFF-006	Duplicate Prevention Validation	High	REQ-SB-106	Duplicate offline records are queued for synchronization	Synchronization executes	The system shall prevent duplicate entries in the backend database and log the reconciliation event
AC-SB-OFF-007	Conflict Resolution Validation	High	REQ-SB-107	A data record is modified locally while offline and the same record is modified in the backend	Synchronization occurs	The system shall apply the documented conflict resolution strategy and maintain a consistent final data state
AC-SB-OFF-008	Local Storage Scope Validation	Medium	REQ-SB-108	Inspection of the local device database	Stored tables and data structures are reviewed	Only approved offline module data shall be present and full backend dataset replication shall not exist
AC-SB-OFF-009	Sync Event Logging	High	REQ-SB-109	A synchronization event occurs	Data is transmitted to the backend	The system shall log timestamp, a non-identifying correlation ID (rotating or session-scoped), and sync status in centralized logging

AC-SB-OFF-010	Sync Failure Notification	Medium	REQ-SB-110	A synchronization attempt fails	The failure is detected	The system shall notify the user within 5 seconds and provide retry instructions
AC-SB-OFF-011	Connectivity-Based Offline Mode	High	REQ-SB-111	Device connectivity is unavailable	The user attempts an offline-supported workflow	The system shall enter offline behavior automatically based on device connectivity status, independent of the manual Offline Mode or Sync toggle
AC-SB-OFF-012	Pause Sync While Online	Medium	REQ-SB-112	Device is online and sync is paused by the user	The user creates an offline-supported record	The record shall be queued locally and not uploaded until sync is resumed
AC-SB-OFF-013	Queue Replay on Reconnect	High	REQ-SB-113	Multiple records exist in the offline sync queue	Connectivity is restored	The system shall replay queued records until completion and mark each record as synchronized or failed with retry status
AC-SB-OFF-014	Deterministic Queue Order and Idempotent Sync	High	REQ-SB-114	Multiple offline-created or offline-modified records exist in the sync queue with timestamps	Connectivity is restored and synchronization begins	The system shall replay queued items in deterministic order (FIFO by timestamp) and repeated submission attempts shall not create duplicate backend records

Table 3.5.2

Infrastructure as Code Acceptance Criteria

AC ID	Title	Priority	Related Requirement	Given	When	Then
AC-SB-INF-001	Infrastructure Provisioning via CloudFormation	High	REQ-SB-201	A CloudFormation template is deployed	Stack execution completes successfully	All defined AWS resources shall be provisioned as specified in the template
AC-SB-INF-002	Version-Controlled Templates	High	REQ-SB-202	Infrastructure templates are updated	Changes are committed	The updates shall be stored in a version-controlled repository with revision history
AC-SB-INF-003	CI/CD Stack Deployment	High	REQ-SB-203	A deployment pipeline is triggered	The CI/CD workflow executes	The CloudFormation stack shall deploy or update automatically without manual intervention
AC-SB-INF-004	Terraform Removal Validation	High	REQ-SB-204	CloudFormation parity has been achieved	Terraform configurations are reviewed	No active Terraform configuration shall provision Team B resources
AC-SB-INF-005	Template Validation Check	High	REQ-SB-207	A deployment pipeline is triggered	Template validation runs	Invalid templates shall fail validation and block deployment
AC-SB-INF-006	Automatic Rollback Confirmation	High	REQ-SB-205	A stack update fails	CloudFormation detects the failure	The stack shall automatically revert to the last stable state

AC-SB-INF-007	Reproducible Deployment Validation	High	REQ-SB-206	A CloudFormation template is deployed in development and production environments	Both deployments complete successfully	Infrastructure resources shall match template definitions and environment parameters without manual modification

Table 3.5.3

Observability Acceptance Criteria

AC ID	Title	Priority	Related Requirement	Given	When	Then
AC-SB-OBS-001	Centralized Logging	High	REQ-SB-301	An application event occurs	The event is processed	The system shall record the event in centralized logging infrastructure
AC-SB-OBS-002	Metrics Collection	High	REQ-SB-302	System operations are running	Performance data is generated	The system shall collect and store defined operational metrics
AC-SB-OBS-003	Distributed Tracing	Medium	REQ-SB-303	A request traverses multiple services	The request is processed end-to-end	The system shall generate trace data linking service interactions
AC-SB-OBS-004	Log Retention Enforcement	High	REQ-SB-304	Operational logs are stored	The retention threshold is reached	Logs shall be retained or archived according to defined retention policies

AC-SB-OBS-005	RBAC Enforcement on Logs	High	REQ-SB-305	A user without administrative privileges attempts to access telemetry logs	Access is requested	The system shall deny access and record the attempt in audit logs
AC-SB-OBS-006	Performance Alert Trigger	High	REQ-SB-306	Defined performance thresholds exist	Monitoring thresholds are breached	The system shall trigger an alert within 60 seconds
AC-SB-OBS-007	Error Rate Monitoring Validation	High	REQ-SB-307	System error rates exceed predefined thresholds	Monitoring detects abnormal activity	An incident log entry shall be created and routed to operations
AC-SB-OBS-008	PHI Exclusion Validation	High	REQ-SB-307	Telemetry logs and monitoring payloads are inspected	Stored log entries and trace metadata are reviewed	No protected health information (PHI) fields shall be present in observability data

Table 3.5.4

Anonymous Feature Usage Analytics Acceptance Criteria

AC ID	Title	Priority	Related Requirement	Given	When	Then

AC-SB-ANA-001	Anonymous Event Logging	High	REQ-SB-401	A feature is used within the application	Usage is recorded	The event shall be logged without storing personally identifiable information (PII) or protected health information (PHI)
AC-SB-ANA-002	Data Minimization Enforcement	High	REQ-SB-402	Analytics data is transmitted	The payload is inspected	No PII or PHI fields shall be included
AC-SB-ANA-003	Aggregated Reporting	Medium	REQ-SB-403	Analytics data is processed	Reports are generated	Outputs shall display aggregated metrics only and shall not identify individual users
AC-SB-ANA-004	Feature Adoption Metric Validation	Medium	REQ-SB-404	Analytics processing is complete	Adoption metrics are calculated	The system shall report feature activation rates as aggregated values

AC-SB-ANA-005	Aggregated Query Access	Medium	REQ-SB-405	Analytics events exist in the telemetry/analytics store and aggregation rules are defined	An authorized administrator requests an analytics summary (e.g., weekly feature counts) using approved backend tooling or database queries	The system shall return only aggregated results and shall not expose user-level identifiers or raw user-specific event payloads
AC-SB-ANA-006	Analytics Opt-Out Validation	Medium	REQ-SB-409	A user opts out of analytics tracking	Analytics events are generated	No usage events shall be transmitted for that user
AC-SB-ANA-007	Analytics Error Logging	Medium	REQ-SB-407	An analytics processing error occurs	The failure is detected	The system shall log the error with timestamp and failure reason
AC-SB-ANA-008	Analytics Data Separation Validation	High	REQ-SB-406	Data storage configurations are inspected	Analytics and clinical repositories are reviewed	Analytics data shall be stored in a logically and physically separate data store from clinical records

AC-SB-ANA-009	Guardrails Enforcement	High	REQ-SB-402	An analytics event contains a blocked key or disallowed event name	The event is processed by the analytics/telemetry layer	The event shall be dropped and the system shall optionally log a non-sensitive debug message in development mode only
AC-SB-ANA-010	Telemetry DB Persistence	High	REQ-SB-408	Telemetry is enabled and an approved event is generated.	The event is processed by the telemetry pipeline	The event shall be stored in the telemetry database table with only approved fields and without PII/PHI, and administrator-accessible reporting outputs shall be aggregated

4. External Interface Requirements

This section specifies all interfaces—human, hardware, software, and communications—as well as operational, reporting, adaptation, and business-rule considerations that govern Team B enhancements and their interaction with the rest of CareConnect and external systems. Team B changes are limited to: offline data storage and synchronization, infrastructure provisioning via AWS CloudFormation, and observability and anonymous feature analytics.

4.1 User Interfaces Overview (Team B Enhancements)

A detailed discussion of CareConnect UI/UX wireframes is maintained in the CareConnect Technical Design Document. Team B does not redesign the user interface; however, Team B enhancements may affect **user-visible behavior** in offline scenarios and may introduce minimal UX elements to explain offline/sync status where appropriate (e.g., "pending sync" indicators). The User Guide documents expected behaviors for offline workflows and syncing.
User interface areas impacted by Team B enhancements include:

- **Mood Tracking / Symptom Tracking / Medication Tracking:** offline entry capture, "pending sync" state, sync completion behavior.
- **EVV:** offline behavior is coordinated with Team A; Team B provides shared offline framework standards but does not implement EVV offline synchronization in this iteration.
- **Calendar:** offline access to essential calendar data and updates that sync on reconnect.
- **Settings / Help (where applicable):** basic explanation of offline behavior and troubleshooting steps.

4.1.1 Offline Status Indicator

This UI element indicates when the device is offline and whether the user has entries waiting to sync. It may appear as a small banner or icon within affected modules to prevent confusion when a server response is not immediately available.

4.1.2 Offline Capability Toggle (Settings)

This UI element allows users to enable or disable offline capability for the application through the Settings page. The toggle provides explicit user control over whether approved modules may store data locally when network connectivity is unavailable.

4.1.3 Pending Sync Marker (Per-Record)

Entries created while offline may display a "pending sync" marker until the system confirms synchronization.

4.1.4 Sync Completion Feedback

After connectivity is restored and synchronization succeeds, the system may remove pending markers and optionally provide brief confirmation in the affected module.

4.2 Hardware Interfaces (Team B)

Team B enhancements use existing mobile hardware and do not introduce new device requirements. Offline capability relies on device storage and (for EVV) may rely on GPS/time services. Interfaces are established through secure connections to CareConnect services when the network is available. **Table 4.2.1B** lists hardware interfaces relevant to Team B enhancements.

Table 4.2.1B

Team B Hardware Interfaces

Subsystem	Device / Platform	Direction	Notes / Protocol
Mobile Client Local Storage	iOS / Android device storage	Internal	Stores offline data in local database; encrypted at rest where supported
Mobile Network Adapter	iOS / Android cellular/Wi-Fi	Outbound/Inbound	Connectivity used for sync, telemetry, and API calls
GPS / Location (EVV)	Mobile device GPS	Internal	Used to capture visit verification context (as allowed by EVV requirements)
Device Clock	Mobile device time services	Internal	Used for ordering offline events and timestamping records

4.3 Software Interfaces (Team B)

Team B enhancements integrate with CareConnect components and approved AWS services. All interfaces enforce secure authentication and encrypted transport. Team B does not modify authentication workflows but relies on existing authentication for authorized API access. **Table 4.3.1-SB** lists software interfaces relevant to Team B enhancements.

Table 4.3.1-SB

Team B Software Interfaces

Req ID	Interface	Type & Protocol	Purpose	Key Requirements
SW-SB-001	CareConnect Backend APIs	REST/HTTPS	Sync offline changes with server	Must support safe retries; must not expose PII/PHI in logs
SW-SB-002	Local Persistence Library (Flutter)	In-app library	Store offline data locally	Must be Flutter-compatible; scoped storage only
SW-SB-003	AWS CloudFormation	AWS API	Provision Team B infrastructure	Templates version-controlled; repeatable stacks
SW-SB-004	Logging/Monitoring Service	HTTPS (AWS or approved tool)	Collect logs/metrics/traces	Must capture operational signals; avoid PII/PHI

SW-SB-005	Analytics Collector	HTTPS	Record anonymous feature usage	Events must be anonymous and separated from ops telemetry

4.4 Communications Interface (Team B)

All communication for Team B features uses secure, encrypted channels. Offline mode must gracefully handle the absence of network connectivity and must retry synchronization later. When online, all traffic is forced over HTTPS/TLS with secure headers and standard retry controls. Latency and retry behaviors align with system Non-Functional Requirements. **Table 4.4.1-SB** lists communications interfaces relevant to Team B enhancements.

Table 4.4.1-SB

Team B Communications Interfaces

Req ID	Channel	Protocol	Port(s)	Encryption	Use
COM-SB-001	Offline Sync API Calls	REST/JSON	443	TLS	Upload offline records; download minimal updates if required
COM-SB-002	Observability Export	HTTPS	443	TLS	Send logs/metrics/traces to monitoring system
COM-SB-003	Anonymous Analytics Events	HTTPS	443	TLS	Send anonymous usage events
COM-SB-004	CloudFormation Provisioning	AWS API	443	TLS	Create/update CloudFormation stacks

4.5 Operations (Team B)

Team B enhancements must support secure and maintainable operations. This includes reliable offline sync, consistent infrastructure provisioning, and operational visibility through monitoring. **Table 4.5.1-SB** lists operational requirements relevant to Team B enhancements.

Table 4.5.1-SB

Team B Operations

Req ID	Area	Requirement
OPS-SB-001	Offline Reliability	Offline records must persist across app restarts and sync when online

OPS-SB-002	Sync Recovery	Sync failures must be retried safely and logged for troubleshooting
OPS-SB-003	Infrastructure Provisioning	CloudFormation templates must support repeatable deployments
OPS-SB-004	Monitoring	Logs, metrics, traces must be available for troubleshooting and performance monitoring
OPS-SB-005	Privacy	Telemetry/analytics must exclude PII/PHI and support audit review where required

4.6 Reporting Requirements (Team B)

Team B introduces reporting requirements mainly for **system health visibility** and **feature usage trends**. These are not patient-facing clinical reports; they are operational and product usage summaries intended for admins, developers, and product stakeholders. **Table 4.6.1-SB** lists reporting requirements relevant to Team B enhancements.

Table 4.6.1-SB

Team B Reporting Requirements

Req ID	Report	Audience	Format	Frequency / Trigger
REP-SB-001	Sync Success/Failure Summary	Admin/Ops	Database query or controlled export (CSV)	Daily/On-demand
REP-SB-002	System Health Metrics	Admin/Ops	Database query or controlled export	Real-time / Daily
REP-SB-003	Anonymous Feature Usage Summary	Product/Stakeholders	Database query, controlled export (CSV), or scheduled summary report	Weekly / On-demand
REP-SB-004	Telemetry Privacy Review Checklist	Security/Privacy Champion	Document	Per release

4.7 Site Adaptations (Team B)

Team B enhancements must support different environments (development, testing, staging, production). Offline behavior should be testable in lower environments without requiring production data. Infrastructure should be reproducible using CloudFormation templates with environment parameters. **Table 4.7-SB** lists site adaptation requirements relevant to Team B enhancements.

Table 4.7-SB

Team B Site Adaptations

Req ID	Environment	Target	Adaptation Actions
SA-SB-001	Development	Local/dev AWS	Use test endpoints; synthetic data; verbose logging
SA-SB-002	Testing/QA	Isolated AWS	Automated sync testing; telemetry validation; privacy checks
SA-SB-003	Staging	Pre-prod AWS	Mirror production configs; validate CloudFormation updates
SA-SB-004	Production	AWS prod	Limit log verbosity; enforce retention; monitor reliability
SA-SB-005	Disaster Recovery	Backup region	CloudFormation supports redeploy; telemetry supports incident review

4.8 Business Rules (Team B)

Team B business rules govern offline scope, infrastructure provisioning standards, and privacy controls for analytics and telemetry. Each rule includes a distinct identifier used for traceability and quality assurance.

Table 4.8-SB

Team B Business Rules

Req ID	Business Rule
BR-SB-001	The system shall allow selected critical features to function when offline by storing required data locally on the user's device.
BR-SB-002	Offline data storage shall be limited to approved modules: mood & symptom tracking and manual calendar.
BR-SB-003	The system shall synchronize locally stored offline data automatically once network connectivity is restored.
BR-SB-004	Local storage shall be limited to essential data required for supported offline modules and shall not replicate the full backend dataset.
BR-SB-005	Team B infrastructure shall be provisioned and managed using AWS CloudFormation as the sole infrastructure-as-code mechanism.
BR-SB-006	Infrastructure definitions shall be maintained in version control to ensure repeatable deployments across environments.
BR-SB-007	The system shall collect operational telemetry (logs, metrics, traces) to support monitoring and troubleshooting.
BR-SB-008	The system shall record anonymous feature usage analytics to measure adoption and frequency without collecting PII/PHI.
BR-SB-009	Telemetry and analytics data collection shall not expose protected health information (PHI) or personally identifiable information (PII).

4.9 Interface-to-Feature Cross-Reference

This section provides bidirectional traceability between external interface requirements defined in Section 4 and the functional features specified in Section 3. This mapping ensures that all interfaces are properly justified by functional needs and that all features have the necessary technical interfaces defined.

Table 4.9

Interface-to-Feature Mapping

Interface ID	Interface Type	Section 3 Feature	Feature ID
SW-SB-001, COM-SB-001	Backend Sync APIs	Offline Data Storage & Synchronization	3.1
SW-SB-002	Local Persistence	Offline Data Storage & Synchronization	3.1
SW-SB-003, COM-SB-004	AWS CloudFormation	Infrastructure Migration	3.2
SW-SB-004, COM-SB-002	Observability Backend (OpenTelemetry Export)	Observability	3.3
SW-SB-005, COM-SB-003	Analytics Service	Anonymous Feature Usage Analytics	3.4
BR-SB-001	Business Rule	Offline Data Storage & Synchronization	3.1
BR-SB-002	Business Rule	Offline Data Storage & Synchronization	3.1
BR-SB-003	Business Rule	Offline Data Storage & Synchronization	3.1
BR-SB-004	Business Rule	Infrastructure Migration	3.2
BR-SB-005	Business Rule	Infrastructure Migration	3.2
BR-SB-006	Business Rule	Observability	3.3
BR-SB-007	Business Rule	Observability	3.3
BR-SB-008	Business Rule	Anonymous Feature Usage Analytics	3.4
BR-SB-009	Business Rule	Anonymous Feature Usage Analytics	3.4

4.10 Use Case Specifications

4.10.1 UC-001: Offline Data Capture and Synchronization

Use Case ID: UC-001
Use Case Name: Offline Data Capture and Synchronization
Primary Actor: Patient
Secondary Actor: Caregiver
Supporting Actor: CareConnect Backend System
Related Requirements (Traceability)

- REQ-SB-101
- REQ-SB-102
- REQ-SB-103
- REQ-SB-104
- REQ-SB-105
- REQ-SB-106
- REQ-SB-107
- REQ-SB-108
- REQ-SB-109
- REQ-SB-110
- REQ-SB-111
- REQ-SB-112
- REQ-SB-113
- REQ-SB-114

Description
This use case describes how users create health-related records while offline and how the system synchronizes those records when connectivity is restored.
Preconditions

- User is authenticated.
- Offline-supported module is accessible.
- Device has no internet connection.
- Offline strategy and data model alignment is maintained across teams, including EVV coordination with Team A.

Trigger
User submits data while offline.
Main Success Scenario

1. User opens approved offline module.
2. Only modules designated for full offline synchronization during this term (Mood & Symptom Tracking and Manual Calendar) proceed through end-to-end synchronization; other modules may store data locally but do not complete synchronization.
3. System detects no connectivity.
4. System allows data entry.
5. User submits record.
6. System encrypts and stores data locally (REQ-SB-101, REQ-SB-103).
7. Record is marked “Pending Sync.”
8. Connectivity is restored (REQ-SB-104).
9. System automatically initiates synchronization (REQ-SB-105).

10. System replays the durable sync queue in deterministic order and uses idempotent submission to prevent duplicates (REQ-SB-113, REQ-SB-114).
11. System prevents duplicate records (REQ-SB-106).
12. Conflicts resolved using defined strategy (REQ-SB-107).
13. Sync event logged (REQ-SB-109).
14. "Pending Sync" indicator removed.

Alternative Flows
A1 – Sync Failure
- Sync fails.
- System retries (REQ-SB-105).
- User notified if failure persists (REQ-SB-110).

A2 – Unauthorized Module
- User attempts unsupported offline module.
- System blocks storage (REQ-SB-102).

Postconditions
- Data synchronized successfully.
- No duplicate entries exist.
- Local storage limited to approved data scope (REQ-SB-108).

4.10.2 UC-002: CloudFormation Infrastructure Deployment

Use Case ID: UC-002
Use Case Name: Infrastructure Provisioning via CloudFormation
Primary Actor: DevOps Engineer
Supporting Actor: AWS CloudFormation
Related Requirements (Traceability)
- REQ-SB-201
- REQ-SB-202
- REQ-SB-203
- REQ-SB-204
- REQ-SB-205
- REQ-SB-206
- REQ-SB-207

Description
This use case describes provisioning Team B infrastructure using AWS CloudFormation.
Preconditions
- CloudFormation templates exist.
- Engineer has AWS permissions.
- CI/CD pipeline configured.

Trigger
Deployment initiated.
Main Success Scenario
1. Engineer triggers CI/CD deployment (REQ-SB-203).

2. Template validation occurs (REQ-SB-207).
3. CloudFormation provisions resources (REQ-SB-201).
4. Templates stored in version control (REQ-SB-202).
5. Stack completes successfully.
6. Infrastructure reproducible across environments (REQ-SB-206).

Alternative Flows

A1 – Stack Failure

- Stack fails.
- CloudFormation rollback initiated (REQ-SB-205).

A2 – Terraform Usage Attempt

- Terraform configuration detected.
- Deployment blocked (REQ-SB-204).

Postconditions

- Infrastructure deployed via CloudFormation only.
- Templates version controlled.

4.10.3 UC-003: Observability Event Logging

Use Case ID: UC-003
Use Case Name: Operational Telemetry Collection
Primary Actor: System Administrator
Supporting Actor: Monitoring Service
Related Requirements (Traceability)

- REQ-SB-301
- REQ-SB-302
- REQ-SB-303
- REQ-SB-304
- REQ-SB-305
- REQ-SB-306
- REQ-SB-307

Description

This use case captures logs, metrics, and traces for system monitoring.

Preconditions

- Backend services operational.
- Monitoring configured.

Trigger

System event occurs.

Main Success Scenario

1. System processes request.
2. Log entry created (REQ-SB-301).
3. Metrics collected (REQ-SB-302).
4. Distributed trace generated (REQ-SB-303).
5. Log retention enforced (REQ-SB-304).

6. Data access restricted via RBAC (REQ-SB-305).
7. Alert generated if threshold exceeded (REQ-SB-306).

Alternative Flows
A1 – PHI Exposure Attempt
- Telemetry includes restricted data.
- System blocks exposure (REQ-SB-307).

Postconditions
- Operational visibility available.
- No PHI/PII present in logs.

4.10.4 UC-004: Anonymous Feature Analytics Collection

Use Case ID: UC-004
Use Case Name: Anonymous Feature Usage Tracking
Primary Actor: System
Secondary Actor: Administrator
Related Requirements (Traceability)
- REQ-SB-401
- REQ-SB-402
- REQ-SB-403
- REQ-SB-404
- REQ-SB-405
- REQ-SB-406
- REQ-SB-407
- REQ-SB-408
- REQ-SB-409

Description
This use case collects anonymous feature usage data while protecting user privacy.
Preconditions
- User has not opted out of analytics.
- The one-time analytics notice has either been acknowledged or recorded as shown.
- Analytics system available.

Trigger
User performs feature action.
Main Success Scenario
1. System generates usage event (REQ-SB-401).
2. Payload validated to exclude PII/PHI (REQ-SB-402).
3. Usage frequency recorded (REQ-SB-403).
4. Adoption metrics calculated (REQ-SB-404).
5. Authorized administrators shall be able to query aggregated analytics data through database queries or backend tooling without requiring a real-time dashboard (REQ-SB-405).
6. Data stored separately from clinical records (REQ-SB-406).

7. The approved event is persisted to the dedicated telemetry database table with only approved fields (REQ-SB-408).
8. Retention policy enforced (REQ-SB-407).

Alternative Flows

A1 – User Opt-Out

- Analytics disabled (REQ-SB-409).
- No events transmitted.

Postconditions

- Anonymous analytics events are stored in the dedicated telemetry table, and administrator-accessible reporting outputs are aggregated.
- No user-identifiable data captured.

5. System Features/Modules

5.1 Offline Data Storage & Synchronization

During this development cycle, full offline synchronization is implemented for Mood & Symptom Tracking and Manual Calendar only. Other modules utilize the shared offline framework but are not fully synchronized.

5.1.1 Offline Data Capture for Mood & Symptom Tracking

5.1.1.1 Description/Priority

This feature allows patients to create and update mood and symptom records when the device has no internet connection. The system stores changes locally on the device and marks them as pending until they can be synchronized to backend services. This prevents data loss and allows users to continue essential tracking even with unreliable connectivity. Priority: High.

5.1.1.2 Stimulus/Response Sequences

Stimulus: User opens the Mood & Symptom module while offline.
Response: The module loads using locally stored data and allows entry creation or updates.
Stimulus: User submits a new symptom entry while offline.
Response: The system stores the entry in the local database and marks it as “Pending Sync.”
Stimulus: User closes and reopens the app while still offline.
Response: The entry remains available and still shows “Pending Sync.”
Stimulus: Network connectivity returns.
Response: The system automatically begins synchronization and removes “Pending Sync” when successfully uploaded.
Stimulus: Synchronization fails (ex: server unreachable).
Response: The system retries later and records a non-sensitive error log entry.

5.1.1.3 Functional Requirements

- REQ-SB-101-1: The system shall store supported care data locally when network connectivity is unavailable.
- REQ-SB-101-2: The system shall encrypt locally stored offline data.
- REQ-SB-101-3: The system shall mark locally stored data as “Pending Sync.”
- REQ-SB-101-4: The system shall prevent full offline synchronization for unsupported modules (for example, Medication Logging) during this development cycle.
- REQ-SB-101-5: The system shall maintain data integrity between offline and online states.

5.1.1.5 Sequence Diagram

Figure 5.1

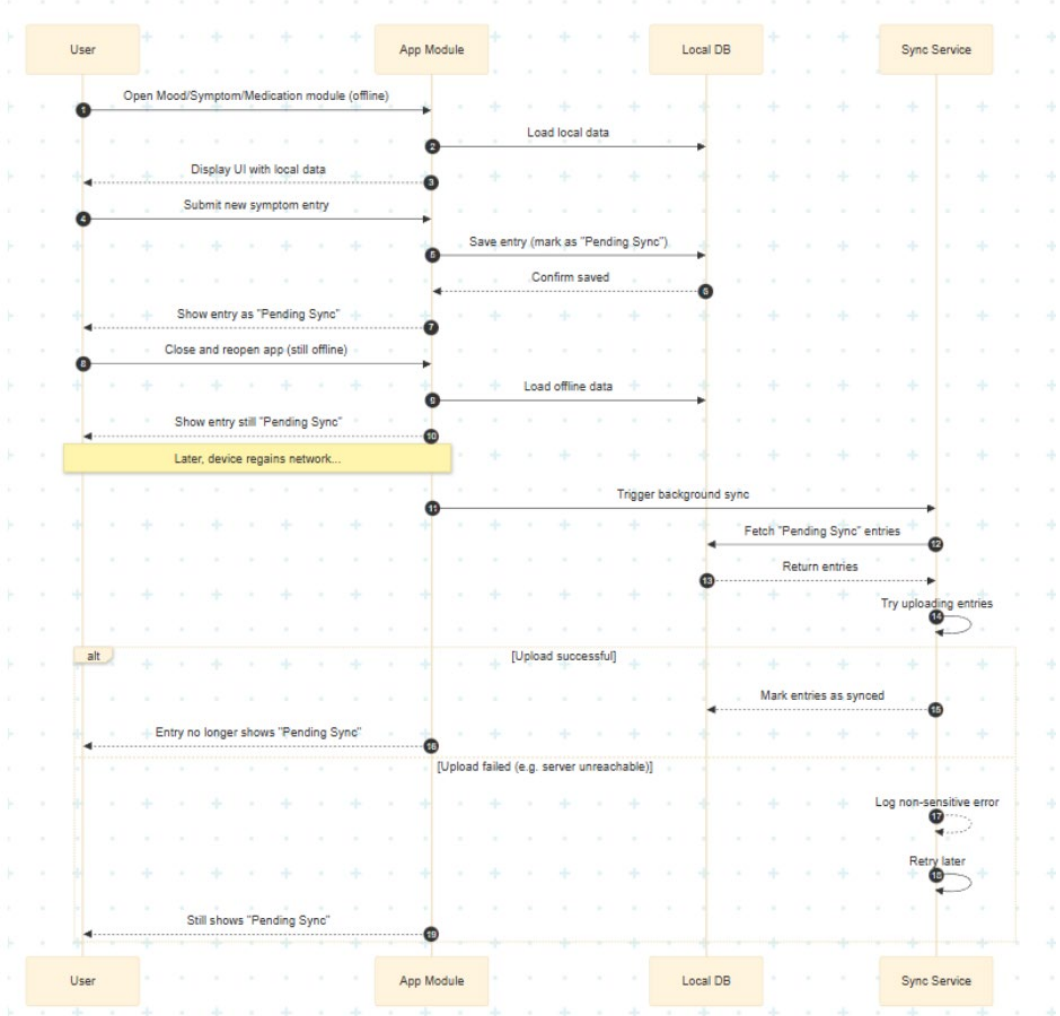


Figure 5.1 depicts the offline data capture and synchronization sequence diagram for mood/symptom/medication tracking.

5.1.2 Offline Calendar Access and Updates (Limited Scope)

5.1.2.1 Description/Priority

This feature supports limited offline access to calendar information and approved offline updates. The goal is to support essential coordination when internet connectivity is unreliable. Local calendar storage is limited to essential fields required for manual calendar entries and does not include full backend replication or automated scheduling data. Priority: Medium–High.

5.1.2.2 Stimulus/Response Sequences

Stimulus: User opens Calendar while offline.
Response: Calendar displays essential locally available data.
Stimulus: User creates an approved event while offline.
Response: Event is stored locally and marked “Pending Sync.”
Stimulus: Connectivity returns.
Response: Offline event is synchronized automatically.

5.1.2.3 Functional Requirements

Synchronization behavior for offline calendar updates shall follow the Synchronization Engine requirements defined in Section 5.1.4 (REQ-SB-105-1 through REQ-SB-109-1).

5.1.2.5 Sequence Diagram
Figure 5.3

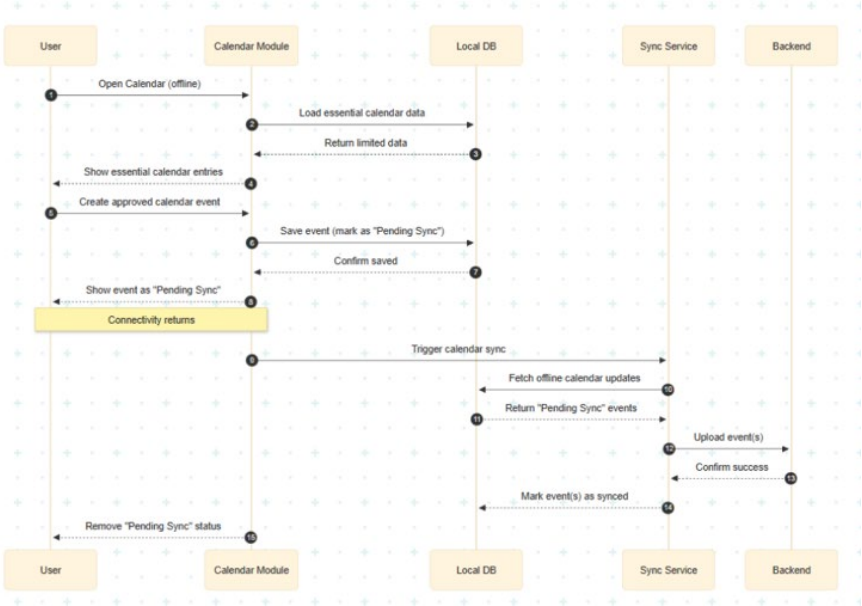


Figure 5.3 depicts the offline calendar access and synchronization sequence diagram.

5.1.3 Synchronization Engine (Retry, Recovery, and Conflict Handling)

5.1.3.1 Description/Priority

This feature defines how offline records are synchronized to backend services once the network becomes available. The sync process must be reliable and must not create duplicates. Conflicts must be handled using a consistent documented strategy. Priority: High.

5.1.3.2 Stimulus/Response Sequences

Stimulus: Connectivity returns.
Response: System automatically detects connectivity and begins sync.
Stimulus: Sync succeeds.
Response: Local records are marked synced and pending indicators are removed.
Stimulus: Sync fails due to temporary server error.
Response: System retries safely with backoff and logs a non-sensitive failure reason.
Stimulus: Conflict occurs (same record changed locally and remotely).
Response: System applies documented conflict strategy and logs a conflict event for support review.

5.1.3.3 Functional Requirements

- REQ-SB-105-1: The system shall automatically initiate synchronization upon network reconnection.
- REQ-SB-106-1: The system shall retry failed synchronization attempts.
- REQ-SB-107-1: The system shall resolve data conflicts using a defined strategy.

- REQ-SB-108-1: The system shall abstract storage logic to prevent backend coupling.
- REQ-SB-109-1: The system shall log synchronization failures for diagnostic purposes.

5.1.3.4 Acceptance Criteria

- **AC-5.1.4-01:** Given a conflict occurs, when synchronization runs, then the system resolves it using the documented strategy without data corruption.

5.1.3.5 Sequence Diagram

Figure 5.4

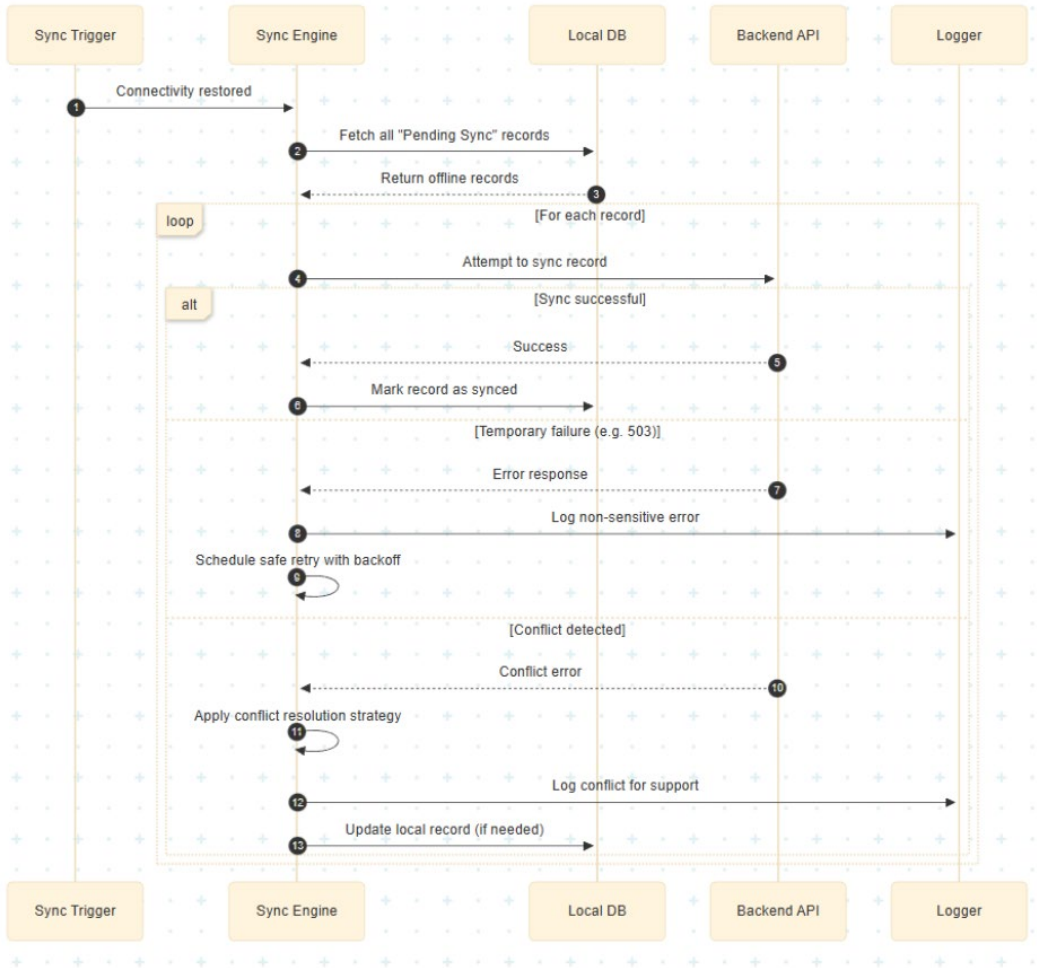


Figure 5.4 depicts the synchronization engine sequence diagram, including retry and conflict handling

5.2 Infrastructure Migration (Terraform → AWS CloudFormation)

5.2.1 CloudFormation Templates for Team B-Owned Infrastructure

5.2.1.1 Description/Priority

This feature ensures Team B infrastructure is provisioned using AWS-native Infrastructure-as-Code through CloudFormation. This reduces fragmentation, improves reproducibility, and ensures future development can deploy environments consistently. Priority: High.

5.2.1.2 Stimulus/Response Sequences

Stimulus: Developer triggers deployment pipeline.
Response: CloudFormation templates are applied to create/update infrastructure.
Stimulus: Stack creation succeeds.
Response: Required resources are created and available.
Stimulus: Stack creation fails.
Response: Deployment is halted and errors are reported to the pipeline output.

5.2.1.3 Functional Requirements

- REQ-SB-201-1: The system infrastructure shall be defined using AWS CloudFormation templates.
- REQ-SB-202-1: CloudFormation templates shall be version controlled.
- REQ-SB-206-1: Deployment shall support rollback on failure.
- REQ-SB-207-1: Infrastructure changes shall be auditable.

5.2.1.4 Sequence Diagram

Figure 5.5

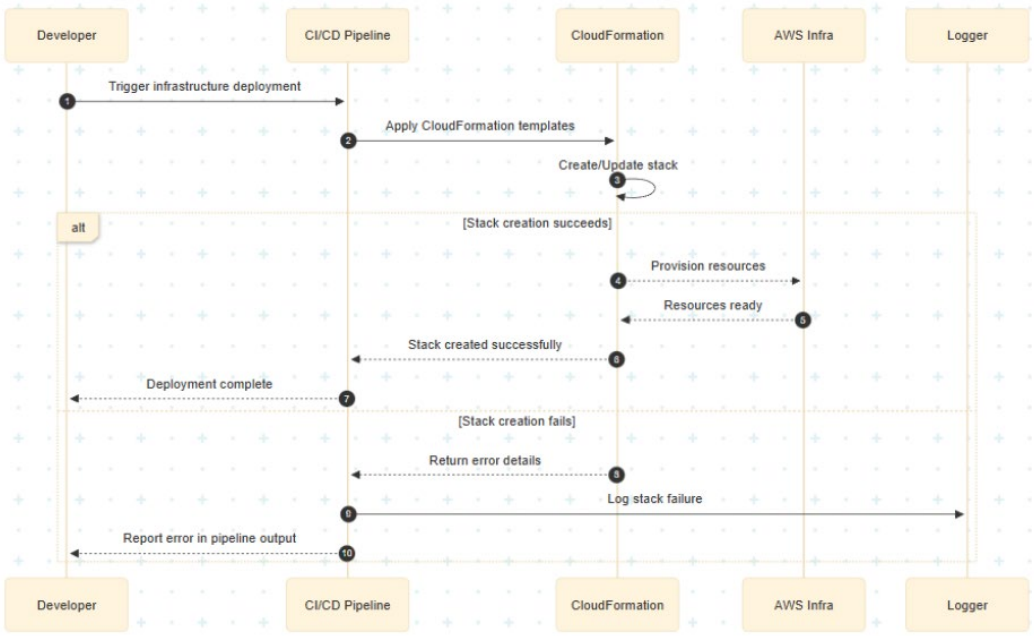


Figure 5.5 depicts the infrastructure provisioning flow using CloudFormation.

5.2.2 Terraform Removal for Team B Infrastructure

5.2.2.1 Description/Priority

After CloudFormation parity is achieved for Team B resources, Terraform must be removed or disabled for Team B scope to avoid multiple provisioning standards. Priority: Medium–High.

5.2.2.2 Stimulus/Response Sequences

Stimulus: CloudFormation templates are validated and deployed successfully.
Response: Terraform scripts for Team B scope are disabled or removed.
Stimulus: New infrastructure change is requested.
Response: The change is implemented only through CloudFormation templates.

5.2.2.3 Functional Requirements

- REQ-SB-204-1: Terraform scripts shall be deprecated.
- REQ-SB-205-1: Terraform dependencies shall be removed from CI/CD pipelines.

5.3 Observability and Anonymous Feature Analytics

5.3.1 Observability (Logs, Metrics, and Traces)

5.3.1.1 Description/Priority

This feature provides operational monitoring to reduce troubleshooting time and improve reliability. Logs, metrics, and traces support diagnosing failures and performance issues. Priority: High.

5.3.1.2 Stimulus/Response Sequences

Stimulus: An error occurs in a backend service.
Response: Error is logged and visible to system operators.
Stimulus: A request latency increases.
Response: Metrics reflect higher latency and can trigger alerts.
Stimulus: A request fails across service boundaries.
Response: A trace is available showing where the failure occurred.

5.3.1.3 Functional Requirements

- REQ-SB-301-1: The system shall generate structured application logs.
- REQ-SB-302-1: The system shall assign unique request identifiers for traceability.
- REQ-SB-303-1: The system shall collect performance metrics including latency and error rates.
- REQ-SB-307-1: The system shall transmit telemetry data securely via HTTPS.

5.3.1.4 Sequence Diagram

Figure 5.6

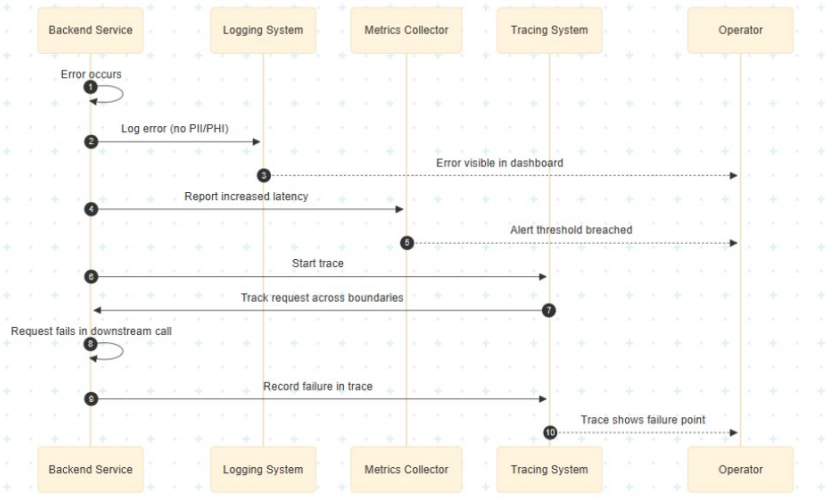


Figure 5.6 depicts the observability pipeline flow (logs/metrics/traces).

5.3.2 Anonymous Feature Analytics (Privacy-Preserving)

5.3.2.1 Description/Priority

This feature provides insight into which features users use and how often. Analytics are anonymous and must not contain PII/PHI. The analytics component is separate from operational telemetry. Anonymous feature analytics are enabled by default and are limited to approved, non-sensitive events. Users are informed of telemetry collection and may opt out at any time through application settings. When analytics are disabled, no usage events are collected or transmitted. Priority: Medium–High.

5.3.2.2 Stimulus/Response Sequences

Stimulus: User completes an action in an offline-enabled module (example: symptom entry created).
Response: System records an anonymous feature event.
Stimulus: Privacy review occurs.
Response: Event payloads can be reviewed and verified to exclude sensitive data.

5.3.2.3 Functional Requirements

- REQ-SB-401-1: The system shall enable anonymous analytics by default and present a one-time notice explaining analytics collection and user control.
- REQ-SB-402-1: The system shall not collect PII or PHI in analytics events, and shall enforce data minimization rules.
- REQ-SB-403-1: The system shall allow users to disable analytics at any time from Settings, and once disabled, the system shall not emit analytics events.
- REQ-SB-405-1: The system shall record feature usage events only when analytics is enabled after applying privacy guardrails.

5.3.2.4 Sequence Diagram

Figure 5.7

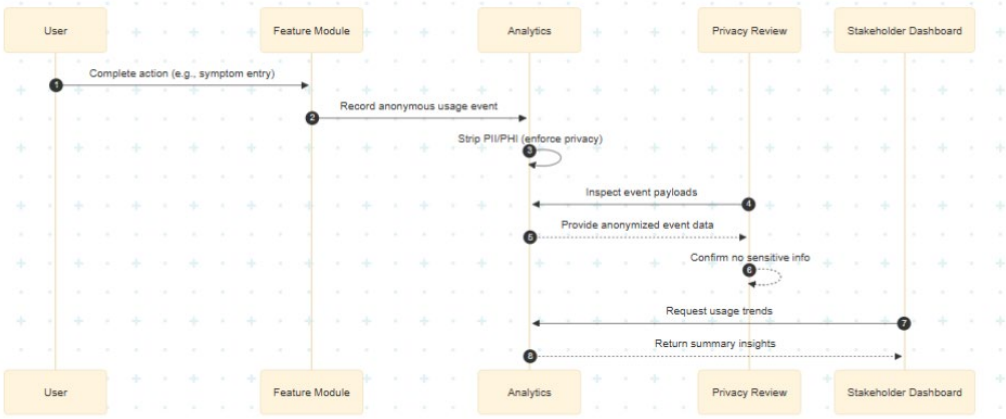


Figure 5.7 depicts the anonymous analytics event flow.

6. Nonfunctional Requirements

These non-functional requirements are scoped to Team B's features and do not replace or supersede existing NFRs defined for the application. For a comprehensive inventory of all nonfunctional requirements applicable to CareConnect, including those outside Team B's scope, refer to the legacy project documentation [4, 9].

6.1 Data Encryption (In-Transit and At-Rest)

To ensure the protection of sensitive personal and healthcare-related data, Team B enhancements must follow encryption requirements for both offline data storage and data synchronization. This is a High Priority requirement because it supports HIPAA/GDPR minimum expectations and reduces risk if a device is lost or compromised.

- NF-6.1.1: All network communication used by offline synchronization, observability telemetry, and analytics shall be encrypted using TLS. (High Priority – supports HIPAA/GDPR compliance)
- NF-6.1.2: Any offline data stored locally for approved modules shall be encrypted using platform-supported mechanisms where available.
- NF-6.1.3: Local data shall not be stored in plain text. (High Priority – protects sensitive user data)

6.2 Regulatory Compliance (HIPAA, GDPR)

Team B enhancements must support privacy and compliance expectations for healthcare-related applications. This is a High Priority requirement and applies especially to telemetry and analytics, since those systems can accidentally capture sensitive information if not controlled.

- NF-6.2.1: Telemetry data (logs, metrics, traces) shall not include any Protected Health

Information (PHI) or Personally Identifiable Information (PII). (High Priority – privacy protection)
- NF-6.2.2: Anonymous feature analytics shall not include identifiers that could reasonably identify an individual (e.g., name, email, phone number, raw message content, or medical text). (High Priority – privacy protection)
- NF-6.2.3: Debugging or operational logging shall avoid recording health-related data entered by users unless explicitly approved. (High Priority – privacy protection)

6.3 Accessibility

Team B enhancements must not reduce accessibility of the existing application. This is a Medium Priority requirement for Team B because Team B is not redesigning UI, but any new or modified indicators (such as "Pending Sync" status) must still be accessible.

- NF-6.3.1: Any new or modified UI indicators (e.g., "Pending Sync" status) shall be readable by screen readers. (Medium Priority – maintain accessibility)
- NF-6.3.2: Status indicators shall not rely solely on color to convey meaning. (Medium Priority – accessibility best practice)

6.4 Offline Function Quality Targets (Reliability and Data Durability)

Offline capability is defined as a functional requirement in the Team B supplemental features section. This subsection defines quality targets that ensure offline functionality is reliable. This is a High Priority requirement because offline support is only useful if users trust that data will not be lost.

- NF-6.4.1: Offline-created records shall persist across app restarts and device sleep/wake cycles. (High Priority – reliability)
- NF-6.4.2: Synchronization shall be resilient to temporary failures and automatically retry without user intervention. (High Priority – reliability)
- NF-6.4.3: The system shall minimize duplicate records caused by reconnect/retry scenarios. (High Priority – data integrity)

6.5 UI/UX Notes (Offline Clarity)

Team B enhancements should prioritize clarity and reduce confusion when users are offline. This is a Medium Priority requirement.

- NF-6.5.1: Users shall be able to record approved offline entries without being blocked. (Medium Priority – usability)
- NF-6.5.2: If an action cannot complete due to lack of connectivity, the app shall display a clear message indicating offline status and expected sync behavior. (Medium Priority – usability) ·
- NF-6.5.3: If "Pending Sync" is displayed, it should be consistent across modules.

6.6 Backup & Disaster Recovery (Team B Scope)

Team B does not own backend backup policies, but Team B enhancements must be designed to support recovery goals and reduce operational risk. This is a High Priority requirement for infrastructure migration and observability.

For Team B scope:

- NF-6.6.1: CloudFormation templates shall support rebuilding Team B infrastructure reliably. (High Priority – operational resilience)
- NF-6.6.2: Observability data shall be sufficient to support troubleshooting and incident review. (High Priority – operational support)
- NF-6.6.3: Offline data on devices shall be treated as temporary and shall not replace backend backups. (High Priority – data safety)

6.7 Performance & Scalability Targets (Team B Scope)

Team B enhancements must not cause noticeable slowdowns or excessive storage usage. This is a High Priority requirement to maintain usability on mobile devices.

- NF-6.7.1: Offline data entry shall be immediate and shall not noticeably delay screen interactions. (High Priority – performance)
- NF-6.7.2: Synchronization shall run in the background without blocking the user. (High Priority – performance)
- NF-6.7.3: Local storage shall remain limited to essential data for approved modules. (High Priority – scalability)
- NF-6.7.4: Observability and analytics event capture shall not noticeably impact app performance or battery life. (High Priority – performance)

6.8 Codebase Normalization and Maintainability (Start Normalization of Codebase)

- NF-6.8.1: New or refactored backend endpoints shall follow the "/v3/api/…" path convention.
- NF-6.8.2: Before introducing new Controllers or DTOs, developers shall review existing implementations and reuse when possible to prevent redundant code.
- NF-6.8.3: New feature directories shall follow the agreed project structure, and teams shall coordinate structural changes with Team B to prevent fragmentation.
- NF-6.8.4: Known legacy issues that are not fixed within the term shall be documented using TODO tags and included in a centralized findings report with mitigation note

Appendix A Requirements Traceability Matrix

Note: Full offline synchronization during this term is limited to Mood & Symptom Tracking and Calendar modules. Other modules leverage the offline framework but are not fully synchronized.

BNS 5 – Offline Data Storage & Synchronization

REQ ID	Description	Child REQ(s)	Use Case	Acceptance Criteria
REQ-SB-101	Offline local storage	101-1 → 101-5	UC-001	AC-SB-OFF-001
REQ-SB-102	Offline module restriction	102-1 → 102-4	UC-001	AC-SB-OFF-002
REQ-SB-103	Local encryption at rest	103-1	UC-001	AC-SB-OFF-003
REQ-SB-104	Connectivity detection	—	UC-001	AC-SB-OFF-004
REQ-SB-105	Automatic synchronization	105-1	UC-001	AC-SB-OFF-005
REQ-SB-106	Duplicate prevention	106-1	UC-001	AC-SB-OFF-006
REQ-SB-107	Conflict resolution strategy	107-1	UC-001	AC-SB-OFF-007
REQ-SB-108	Limited local dataset scope	108-1	UC-001	AC-SB-OFF-008
REQ-SB-109	Sync event logging	109-1	UC-001	AC-SB-OFF-009
REQ-SB-110	Sync failure notification	—	UC-001	AC-SB-OFF-010
REQ-SB-111	Connectivity-based offline mode	—	UC-001	AC-SB-OFF-011
REQ-SB-112	Pause sync while online	—	UC-001	AC-SB-OFF-012
REQ-SB-113	Durable sync queue replay on reconnect	—	UC-001	AC-SB-OFF-013
REQ-SB-114	Deterministic queue order + idempotent submission	—	UC-001	AC-SB-OFF-014

BNS 6 – Infrastructure Migration (CloudFormation)

REQ ID	Description	Child REQ(s)	Use Case	Acceptance Criteria
REQ-SB-201	Provision via CloudFormation	201-1	UC-002	AC-SB-INF-001
REQ-SB-202	Version-controlled templates	202-1	UC-002	AC-SB-INF-002

REQ-SB-203	CI/CD stack deployment	—	UC-002	AC-SB-INF-003
REQ-SB-204	Remove Terraform configs	204-1	UC-002	AC-SB-INF-004
REQ-SB-205	Stack rollback support	205-1	UC-002	AC-SB-INF-006
REQ-SB-206	Reproducible environments	206-1	UC-002	AC-SB-INF-007
REQ-SB-207	Template validation	207-1	UC-002	AC-SB-INF-005

BNS 7 – Observability (Logs, Metrics, Traces)

REQ ID	Description	Child REQ(s)	Use Case	Acceptance Criteria
REQ-SB-301	Centralized logging	301-1	UC-003	AC-SB-OBS-001
REQ-SB-302	Performance metrics collection	302-1	UC-003	AC-SB-OBS-002
REQ-SB-303	Distributed tracing	303-1	UC-003	AC-SB-OBS-003
REQ-SB-304	Log retention policy	—	UC-003	AC-SB-OBS-004
REQ-SB-305	RBAC for telemetry	—	UC-003	AC-SB-OBS-005
REQ-SB-306	Alert threshold monitoring	—	UC-003	AC-SB-OBS-006
REQ-SB-307	PHI exclusion enforcement	307-1	UC-003	AC-SB-OBS-007, AC-SB-OBS-008

BNS 7 – Anonymous Feature Analytics

REQ ID	Description	Child REQ(s)	Use Case	Acceptance Criteria
REQ-SB-401	Anonymous usage collection	401-1	UC-004	AC-SB-ANA-001
REQ-SB-402	No PII/PHI in analytics	402-1	UC-004	AC-SB-ANA-002
REQ-SB-403	Usage frequency tracking	403-1	UC-004	AC-SB-ANA-003
REQ-SB-404	Feature adoption metrics	—	UC-004	AC-SB-ANA-004
REQ-SB-405	Aggregated analytics query access	405-1	UC-004	AC-SB-ANA-005
REQ-SB-406	Separate analytics storage	—	UC-004	AC-SB-ANA-008

REQ-SB-407	Configurable retention policies	—	UC-004	AC-SB-ANA-007
REQ-SB-408	Telemetry DB persistence (separate table)	—	UC-004	AC-SB-ANA-010
REQ-SB-409	User opt-out of analytics	—	UC-004	AC-SB-ANA-006

Software Requirements Document

for

CareConnect

Version 6

Prepared by: SWEN 670 Class of Spring 2026 Team C

University of Maryland Global Campus

January 26th, 2026

Revision History

Team Name	Date	Reason for Changes	Version

CareConnect Team C Spring 2026	01/24/2026	Initial document submission.	1.0
CareConnect Team C Spring 2026	02/07/2026	Discovery Updates Milestone 2	2.0

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the functional and non-functional requirements of CareConnect, a HIPAA/GDPR compliant web application that connects patients, care givers, and their healthcare team. This SRS will communicate with the development team and relevant stakeholders about the design, development techniques, testing, and metrics for acceptance of the CareConnect application.

Purpose: Define functional & non-functional requirements for the CareConnect application.

Audience: Clients, Stakeholders, professor, mentors, development team.

Scope: Patient-caregiver application for managing scheduling, visit notes, and billing.

1.2 Document Conventions (Definitions, Acronyms, and Abbreviations)

The definitions, acronyms, and abbreviations used in this document are specified in Table 1 & 2. Some acronyms and definitions were inherited from CareConnect SRS, Aug 30, 2025.

Table 1 Document Conventions

Term	Definition
AI (Artificial Intelligence)	A system capable of doing human like tasks involving pattern recognition, speech recognition, generating content
Amazon Web Services (AWS)	A suite of different cloud services offered by amazon including different database and server services needed for CareConnect.
Android OS	An open-source mobile operating system developed by Google.
App Runner	A serverless environment to run applications offered by AWS
CareConnect	A web/mobile application developed by SWEN 670 cohorts to assist caregivers and patients with healthcare management.

Caregiver	An individual who assists with the care of an individual with some extraordinary needs.
Care Receiver	An individual with some extraordinary healthcare needs.
Dart Tool	Programming language and toolset used to create Flutter applications.
Fargate	A serverless environment to run applications offered by AWS
Flutter	An open-source UI software development kit (SDK) created by Google, capable of building natively compiled applications for mobile, web, and desktop from a single codebase.
HIPAA (Health Insurance Portability and Accountability Act)	U.S. regulation intended to protect an individual's healthcare related information
Roll based Access Control	A way of controlling which screens and data a user has access to according to the roll they are assigned in the system.
iOS	A mobile operating system developed by Apple for iPhone
Unit testing	Refers to automated testing used in software development to help catch errors in compile time.
User Interface (UI)	The visual and interactive components of the application which allows users (Caregivers and receivers) to interact with the system.
User Experience (UX)	The overall experience and satisfaction a user has when interacting with the application, including ease of use, efficiency, and accessibility.

(As cited in the CareConnect SRS, Aug 30, 2025)

Table 2 Acronyms

Acronyms	Definitions
AC-###	Acceptance criterion tested for a requirement/use case
AI	Artificial Intelligence
API	Application Programming Interface
App	Application (mobile or web-based software)
AWS	Amazon Web Services
BNS	Business Need Statement
DB	Database
GUI	Graphical User Interface
HIPAA/GDPR	U.S. healthcare privacy law / EU privacy regulation
IoT	Internet of Things
PHI/PII	Protected Health Information / Personally Identifiable Information
PMP	Project Management Plan
QA	Quality Assurance
RAM	Random Access Memory
RBAC	Role-Based Access Control
RDS	Relational Database Service
REQ-###	Requirement ID
REST	Representational State Transfer (API standard)

SRS	Software Requirements Specification
TC-###	Test case that verifies an AC
UC-###	Use case
UI	User Interface
UX	User Experience
SLA/SLO	Service Level Agreement/Objectives
MA	Medicaid Number
EOR	Employer of Record

(As cited in the CareConnect SRS, Aug 30, 2025)

1.3 Intended Audience and Reading Suggestions

This SRS is intended for all project stakeholders, professors, mentors, and the development team. This SRS along with the project management plan (PMP) together will describe the foundation of the requirements Team C has been commissioned to complete and how the team plans to achieve and measure success.

1.4 Product Scope

The scope of this software development effort is to augment the existing CareConnect application with three distinct new requirements for the application. The requirements, migrate to Fargate or App Runner, design and implement new unit testing paradigm, and redesign and implement a new role-based access control system are spelled out in the following sections of this SRS.

1.4.1 In-Scope

Features, capabilities, and services that are included in the current CareConnect release.

1.4.1.1 Migration to Fargate/App Runner

- Investigate which AWS service from the short list (Fargate, and App Runner) is the better candidate to host CareConnect and operates within any maintenance or cost constraints.
- Identify what steps must be taken to deploy CareConnect to the chosen host.
- Execute identified steps and deploy CareConnect to new host service.

1.4.1.2 Design and Implement New Unit Testing Paradigm

- Identify any pain points with the current unit testing design.
- Create a new streamlined testing paradigm with functions identified for automated testing and user testing.
- Implement new testing paradigm for the existing CareConnect Application

1.4.1.3 Redesign and Implement Roll Based Access Control

- Identify current roll-based access control system, what can be reused, and what needs to be removed.
- Design a new streamlined roll-based access control system.
- Ensure new system redirects users away from screens they are not supposed to have access to or provide a helpful message with a clear exit route.
- Ensure new system locks down API endpoints so in the event a user does land somewhere they are not supposed to be, they do not gain access to unauthorized data.

1.4.2 Out-of-Scope

Features and capabilities that are explicitly excluded from the current release or deferred to future versions.

1.4.2.1 Investigate Docker Stack for Production Deployment

- Investigating containerizing the entire application (front end, back end, and database) for easier updating and potentially removing RDS costs.

1.4.2.2 Unit Testing of New Features

- Implementing unit or user testing for any new functionality being developed by Teams from the Spring 2026 cohort.

1.4.2.3 Implement New Role-Based Access Control for New Pages and API Endpoints

- Implement the new role-based access controls for any new pages, feature, or api endpoints being developed by another team in this Spring 2026 cohort.

1.5 References

- Requirements document provided by stakeholder mentors.

- Software Requirements Specification document from Summer/Fall 2025 cohort https://umgc-cappms.azurewebsites.net/download/d342ee6d-bbe6-49ee-aa03-75a359b26bf8----SWEN670_CareConnect_Fall2025_SRS.pdf

---

- AWS Fargate https://aws.amazon.com/fargate/
- AWS App Runner https://aws.amazon.com/apprunner/
- An introduction to unit testing https://docs.flutter.dev/cookbook/testing/unit/introduction
- JUnit User Guide https://docs.junit.org/6.0.2/overview.html

2. Overall Description

2.1 Product Perspective

CareConnect is a secure, cross-platform care system designed to support collaboration between patients and their caregivers. The system serves as a centralized digital hub for managing health-related activities, communication, and care oversight in home-based and remote care environments.

The application is envisioned as a platform that integrates core caregiving workflows such as communication, scheduling, and monitoring. From a system architecture standpoint, CareConnect is composed of a client-facing application layer (web and/or mobile), backend services that expose secured APIs, and cloud-based infrastructure responsible for hosting, deployment, and data persistence. The application operates as a standalone system but is designed to support integration with external services and future system expansion, as defined in prior versions of the SRS.

The Spring 2026 v1 release does not represent a new product or major functional expansion. Instead, it focuses on augmenting and strengthening the existing CareConnect platform by improving how the system is deployed, tested, and secured. These improvements are intended to enhance system maintainability, scalability, and governance while preserving all existing user-facing functionality.

Specifically, this release:

- Re-evaluates the application's cloud hosting strategy within AWS
- Introduces a redesigned unit testing approach to improve software quality
- Refactors the role-based access control mechanism to enforce authorization more consistently across the user interface and backend APIs

CareConnect's conceptual role within the broader care ecosystem remains unchanged. The system continues to serve as a digital bridge between caregivers and patients, supporting coordination and oversight in home-based and remote care scenarios. The enhancements introduced in Spring 2026 provide a more robust technical foundation upon which future functional growth can be built.

2.2 Product Features

Table 3 Core Product Features

Category	Feature	Description	User Role(s)
Onboarding & Authentication	Login & Registration	Email/password, SMS-OTP, SSO support	Care Recipient, Caregiver
	User Registration	Sign-up via email and password.	Care Recipient, Caregiver
	Password Reset	Password recovery via email.	Care Recipient, Caregiver
Billing & Subscription	Stripe Integration	Secure billing, plan management	Caregiver
	Subscription Management	View, activate, or cancel monthly billing plans.	Caregiver
	Transparent Pricing Display	Shows payment plan.	Caregiver
Scheduling & Notifications	Care Scheduling	Set up custom schedules for meals, etc.	Caregiver
	Reminder Notification	Task notification to Care Recipients and caregivers.	Care Recipient, Caregiver
	Shared Calendar	View/manage tasks, reminders, caregiver handoffs	Care Recipient, Caregiver
	Virtual Check-ins	Remote check in on Care Recipient for daily monitoring, wellness check	Caregiver

Gamification	Compliance Rewards	Earn rewards (star, badge) on completing the task	Caregiver, Patient
	Leaderboard	Caregiver Ranking	Caregiver
	Motivational Feedback	Positive reinforcement	Patient
Analytics & Reporting	Health Dashboards	Graphs/summaries of symptoms logs, meds, meals, etc.	Caregiver
	Adherence Monitoring	Track completed task vs missed task	Caregiver
	Alert Logs	List of high-priority alerts rigged by patient inputs	Caregiver
Security & Compliance	HIPAA/GDPR Compliance	Protect user data during storage and transmission.	All users (involved)
Device and Third-party Integration (Future Not included in MVP)	Wearable Device	Apple Health/Google Fit	Caregiver
	Smart home (Future enhancement)	Nest/Ring/RTSP camera integration for fall detection, live view	Caregiver
	Telehealth (Future enhancement)	Zoom/Teams link for online doctor visits.	Caregiver, Care Recipient
Communication	Telemedicine & Visit Notetaker	Voice diarization, trigger words, searchable notes	Care Recipient, Caregiver
	Voice & Video Integration	The system will support audio and video calling features between caregivers and patient	Caregiver, Care Recipient
AI & Analytics	AI integration with Vital Anomaly Detection	Caregivers will receive alerts for anomalies (e.g. elevated heart rate)	Caregiver

Accessibility	ADA/ASL Features	Screen reader, color contrast, text-to-ASL support	Care Recipient, Caregiver

(Table 2: as cited in the CareConnect SRS, Aug 30, 2025)

2.3 User Classes and Characteristics

Table 4 User Classes and Characteristics

User Class	Role Description	Core Responsibilities	Technical Skill	Access
Caregiver	Manages care schedules, monitoring, billing	Add patients, create tasks, respond to alerts	Medium–High	Full
Care Reciever	Care recipient	Log symptoms, view schedules, respond to reminders, SOS alerts	Low–Medium	Limited
Family Member	Read-only support role	View reports, monitor logs, receive alerts	Low	Read-only
Admin (internal)	Backend/ system oversight	Manage logs, compliance, permissions	High	Full

(as cited in the CareConnect SRS, Aug 30, 2025)

2.4 Operating Environment

CareConnect is intended to operate in a modern, internet-connected environment and will be accessible via:

- Mobile devices (iOS and Android platforms)
- Web browsers on desktop and tablet devices

The system relies on cloud-based infrastructure to support availability, scalability, and secure data storage. All user interactions occur over encrypted network connections to ensure confidentiality and integrity of data.

2.4.1 Hardware Requirements

For the mobile application the user must have a tablet or smart phone with an ARM64 processor and at least 2GB of RAM, a camera and microphone.

CareConnect also supports wearables integration with Fitbit using Fitbit web APIs. For this functionality, the user must use a Fitbit and it can track heart rate, steps, and SpO2 data (CareConnect SRS, Aug 30, 2025 was referenced in this section).

2.4.2 Software RequirementsThe user must use a device with Android 11 or higher and IOS 15 or higher. The UI was created using the Flutter SDK and dart programming language for cross platform functionality. Other third-party integrations include Stripe, AWS, Auth 2.0, USPS digest API, Google Calendar, CodeMagic, GitHub Actions, Terraform, and OAuth 2.0 (CareConnect SRS, Aug 30, 2025 was referenced in this section).

2.4.3 Network Requirements

For video calling an internet speed of at least 1Mbps is recommended to support uploads, API communication and video. A maximum latency of 300ms is required for real-time features like notifications and calls. Finally, a stable internet connection is required for syncing data, receiving notifications, and calls (CareConnect SRS, Aug 30, 2025 was referenced in this section).

2.4.4 System Environment

as cited in the CareConnect SRS, Aug 30, 2025...

- AWS serverless-first deployment with scale-to-zero principles.
- TLS 1.3 encryption, HIPAA-eligible services (Cognito, RDS, S3).
- Monitoring: AWS CloudWatch + Lake Formation for immutable audit trails.
- Auto-scaling for resilience and cost efficiency.
- All commits must be automatically deployed to an isolated AWS test environment for integration and compliance testing prior to advancement.

2.4.5 Environmental Constraints

as cited in the CareConnect SRS, Aug 30, 2025...

- Partial offline functionality is available for caregivers to access cached patient data, schedules, and care instructions.
- UI/UX design must account for varying screen sizes, limited bandwidth, and potentially low device performance on older phones.

- Basic accessibility features such as screen reader compatibility and voice input are considered, but full compliance with accessibility standards is deferred to future releases; full WCAG 2.1 Level AA deferred.

2.5 Design and Implementation Constraints

The design and implementation of the Spring 2026 v1 release of CareConnect are governed by the following constraints, which intentionally limit the scope of development to infrastructure, testing, and access control enhancements for the existing application.

Functional Scope Constraint

No new user-facing features, workflows, pages, or business logic shall be introduced as part of this release. All functional product features described in prior versions of the CareConnect SRS are carried forward without modification. Any new functionality developed by other teams during the Spring 2026 term is explicitly excluded from this SRS and shall not be incorporated into the work defined herein.

Deployment and Infrastructure Constraint

The scope of deployment-related work is restricted to evaluating and migrating CareConnect to a managed AWS hosting service selected from the approved shortlist (AWS Fargate or AWS App Runner). Broader infrastructure changes—including full application containerization, database re-platforming, or replacement of existing persistence mechanisms—are outside the scope of this release.

Testing Scope Constraint

Testing activities are limited to the redesign and implementation of a unit testing paradigm for existing CareConnect functionality. The development of test coverage for newly introduced features, experimental modules, or functionality created by other Spring 2026 teams is explicitly excluded.

Access Control Constraint

Role-based access control (RBAC) improvements are limited to refactoring and enforcing authorization rules for existing roles, screens, and API endpoints. The introduction of new roles, permissions, or authorization logic for newly developed pages, features, or services is outside the scope of this release.

Inter-Team Dependency Constraint

This SRS assumes parallel development efforts may occur during the Spring 2026 term. Dependencies on deliverables from other teams are minimized, and no assumptions are made regarding integration with newly developed components outside the scope of this document.

2.5.1 Standards Compliance

CareConnect's architecture adheres to established industry standards governing data security, usability, and regulatory alignment.

HIPAA requirements are enforced to protect all Protected Health Information (PHI). GDPR provisions are also implemented, supporting data portability, consent management, and erasure rights. The MVP targets partial WCAG 2.1 Level AA conformance; full ADA alignment, including specialized integrations such as the USPS Digest, is scheduled for post-MVP releases.

Although FHIR/HL7 interoperability is not currently required, it will be evaluated for support of future Electronic Health Record (EHR) integrations.

(as cited in the CareConnect SRS, Aug 30, 2025)

2.5.2 HIPAA Compliance

CareConnect employs a defense-in-depth strategy to satisfy HIPAA's privacy and security mandates for PHI through technical and procedural safeguards.

The application is deployed exclusively on HIPAA-eligible AWS services, including Amazon S3 and Amazon Cognito, configured with strict access restrictions and encryption protocols. PHI is encrypted in transit using TLS 1.3 and at rest using AWS KMS-managed keys (AES-256), with this protection extending to all media files and system logs.

A Role-Based Access Control (RBAC) model enforces the principle of least privilege so only authorized personnel (such as caregivers and administrators) can access PHI. All system-level actions, authentication events, and PHI access attempts are recorded in immutable logs via AWS Lake Formation to facilitate compliance audits and security monitoring.

Internal administrative access requires hardware-backed Multi-Factor Authentication (MFA), specifically FIDO2 or U2F security keys. To comply with jurisdictional laws, PHI is stored solely in U.S.-based data centers. Disaster recovery protocols include automated lifecycle management and a 35-day backup retention window.

(as cited in the CareConnect SRS, Aug 30, 2025)

2.5.3 Hardware Limitations

The system design accounts for performance constraints inherent to mobile and IoT hardware environments.

Client devices must meet minimum specifications: an ARM64 processor, 2 GB of RAM, and functional integrated audio/visual input (camera/microphone). The MVP supports the Fitbit Web API, with real-time analytics processed off-device where possible to reduce battery drain and accommodate processing limitations on the wearable itself. Smart home integration is currently restricted to the Google Nest API; support for other ecosystems (such as Ring and Arlo) is deferred to future releases.

To prevent caregivers from becoming overwhelmed by excessive notifications, the system employs configurable thresholds that balance timely updates against the risk of oversaturation.

(as cited in the CareConnect SRS, Aug 30, 2025)

2.6 User Documentation

CareConnect will be available for multiple platforms, as discussed above. Therefore, to help our users to get the most out of our solution, we will provide documentation in multiple formats. In-App help and support that is a direct on-hand way of presenting and making documentation accessible. It comes with a set of pages directly accessible from the mobile and the web version. It involves the implementation of tooltips on components to give the user real-time learning experience.

This subsection describes documentation strategies for supporting diverse users, ranging from patients with limited technical literacy to administrators with advanced system oversight responsibilities. Documentation will support all user classes and conform to ADA accessibility requirements.

- Tooltips, walkthroughs, and contextual help.
- Captioned and transcribed short tutorials for onboarding and key features.
- Web-based repository for advanced features, compliance workflows, and admin functions.
- Decision Trackers will be included in this document and other deliverables.
- End-to-end screen mockups (Figma/FlutterFlow) will be used to provide
- authoritative workflow documentation. (as cited in the CareConnect SRS, Aug 30, 2025)

2.7 Assumptions

The following assumptions apply to the Spring 2026 v1 release of CareConnect and define the conditions under which the requirements in this SRS are considered valid:

- CareConnect is an existing, functioning application with established user-facing features as defined in prior SRS versions, and those features will remain functionally unchanged during this release.
- The primary objective of the Spring 2026 effort is to enhance non-functional aspects of the system, specifically application deployment, unit testing, and role-based access control enforcement.
- No new product features, user workflows, pages, or business logic are expected to be introduced by the team responsible for this SRS.
- Existing user roles and high-level authorization concepts are assumed to be present and stable, serving as the baseline for RBAC refinement rather than expansion.
- The CareConnect codebase, deployment artifacts, and documentation are assumed to be sufficiently complete to support analysis, migration, testing, and access control refactoring.
- Users will continue to access CareConnect through supported platforms and devices without requiring changes to client-side hardware or operating systems.
- Parallel development efforts by other teams may occur during the Spring 2026 term; however, their deliverables are assumed to be independent of the work defined in this document.

2.8 Dependencies

The successful implementation of the Spring 2026 v1 release is dependent on the following external and internal factors:

- Availability of the existing CareConnect source code, including backend services, user interface components, and current access control logic.
- Access to appropriate AWS resources, accounts, and permissions to evaluate and deploy the application using AWS Fargate or AWS App Runner.
- Stability of existing application features during the migration, testing, and RBAC refactoring processes.
- Continued availability of current third-party services and integrations already used by CareConnect, without requiring modification or reconfiguration as part of this release.
- Cooperation from stakeholders to provide clarification on existing deployment, testing, and authorization mechanisms as needed.

- Dependencies on other Spring 2026 teams are intentionally minimized. Integration with new pages, features, APIs, or services developed outside the scope of this SRS is not required and is explicitly excluded.

2.9 Constraints

The following constraints have been identified in the development of CareConnect:

- The entire development cycle for a minimal viable product must be complete within a 12-week period. This includes a fully completed product. A beta version must be ready within a 4-week time period.
- Only open-source and free tools must be leveraged. There is no budget to adopt any solutions that may incur a cost.
- A scale-to-zero and serverless architecture must be prioritized.
- Full regulatory compliance may not be initially met by the length of time as the verification process would be referred to an external division.
- CareConnect must be developed within the constraints of the chosen technology stack.
- Complete conformity to accessibility standards will be limited for CareConnect in this term due to time constraints.
- managed/cloud-first; MVP excludes email/SMS and full accessibility.

This subsection details the non-negotiable limitations on system development, ranging from regulatory compliance to technical restrictions. These constraints define the project boundaries and inform trade-off decisions.

- Development is limited to a 12-week cycle, with beta testing by Week 4.
- Budget constraints mandate open-source + AWS credit-eligible services.
- Architecture must use serverless-first and scale-to-zero principles.
- HIPAA/GDPR compliance is mandatory; full accessibility deferred.
- Hardware minimum: ARM64, 2 GB RAM.
- CI/CD pipelines with continuous deployment to an isolated test environment are unavoidable.
- Wearable integration is limited to Fitbit at MVP.
- Smart home integration is limited to Nest due to API availability.
- Secure storage of telemedicine notes/transcripts is required.
- Notification systems must prevent alert fatigue for caregivers.

(as cited in the CareConnect SRS, Aug 30, 2025)

3. System Features

This section describes the system-level features addressed in the Spring 2026 v1 release of CareConnect. The features defined in this section represent non-functional, architectural, and governance-oriented system enhancements that support the existing CareConnect feature set.

No new end-user workflows, screens, or business capabilities are introduced as part of this release. All system features described below are intended to improve the reliability, security, maintainability, and operational readiness of the CareConnect platform while preserving existing functional behavior.

3.1 Application Hosting and Deployment Modernization

3.1.1 Description

CareConnect shall support deployment using a managed AWS hosting service selected from AWS Fargate or AWS App Runner. This system feature governs the operational environment of the application and does not alter existing business logic or user workflows.

The purpose of this feature is to ensure that CareConnect operates within acceptable operational, maintenance, and cost constraints while supporting scalable and reliable deployment.

For this requirement Team C has chosen to target Fargate for the new production host. The decision came down to three criteria; cost, flexibility, and level of effort to maintain then balanced the benefits and drawbacks of each. For cost Fargate estimates were about one half the cost of App Runner (~$0.04048/hr vs. ~$0.07800/hr. Flexibility was another plus for Fargate as CareConnect would have more control over the container itself as App Runner only gives enough control to run an application on a container with no control over the container. Because of the lower cost and more flexibility Fargate does have a higher level of effort which is a drawback however team C believes that the cost of App Runner alone takes it out of the running.

The current CareConnect system baseline utilizes AWS lambda for production hosting. The most significant problem with AWS lambda is that it doesn’t support all the necessary packages used in creating the backend part of the application. The goal of moving to a container environment

is to gain that flexibility. Additional benefits and considerations for the future could be to do a cost comparison of RDS and running an additional instance of Fargate to house a production level instance of PostgreSQL for possible cost savings and service consolidation.

Requirement ID	Requirement Description
REQ-1	Migrate production environment to Fargate
REQ-1.1	Build out docker file for spring boot backend
REQ-1.2	Deploy spring boot backend to Fargate

3.1.2 Functional Behavior (System-Level)

- The system shall support deployment of the existing CareConnect application to the selected AWS hosting service.
- The system shall remain accessible to users following migration without requiring changes to existing user workflows.
- The system shall support repeatable deployment processes suitable for future releases.

3.1.3 Constraints and Limitations

- This feature does not include redesigning the application architecture or business logic.
- Full application containerization, database re-platforming, or infrastructure redesign beyond the selected hosting service is excluded.
- Deployment changes shall not introduce new application features or modify existing functionality.

3.1.4 Acceptance Criteria

The application hosting and deployment modernization requirement will be accepted as complete when the production back end is running in a Fargate instance with all previous features functional and no loss of performance. The end user should not notice any difference with this change.

Hosting Functional Acceptance Criteria (Functional Focus)

AC ID	**Criterion**	**Verification**	**Pass Condition**
AC-3.1.4-001	Successful Fargate Deployment – The backend shall be deployed to AWS	Automated health checks and API tests	All endpoints return expected HTTP status codes

	Fargate with all endpoints operational		
AC-3.1.4-002	Zero Functional Regression – Existing features shall function identically pre- and post-migration	Smoke testing of MVP features	No critical or high-severity defects introduced
AC-3.1.4-004	Repeatable Deployment Process – Deployment shall be automated via CI/CD	Execute deployment pipeline multiple times	3 consecutive successful deployments
AC-3.1.4-005	User Experience Unchanged – End users shall observe no change in behavior or availability	User Acceptance Testing (UAT)	100% stakeholder approval
AC-3.1.4-008	Docker Container Validation – Backend container shall build and pass security scans	Docker build and ECR scan	Successful build with zero critical vulnerabilities

3.2 Quality Assurance and JUnit Testing

3.2.1 Description

CareConnect shall incorporate a redesigned Junit testing paradigm focused on improving test structure, coverage, and maintainability for existing system functionality. This feature addresses how the system is validated rather than how it behaves from a user perspective.

The objective of this feature is to provide a consistent and repeatable testing approach that supports regression detection and long-term maintainability.

The current CareConnect system baseline for unit testing is very limited code coverage, and to package the spring boot application, unit tests must be skipped all together. Because mock data wasn't implemented previous development efforts relied on attempting to invoke external AWS

services which caused them to fail. Since unit testing is skipped all together because of these issues, code coverage can be considered at zero.

Requirement ID	Requirement Description
REQ-2	Finish implementing unit testing on existing code base
REQ-2.1	Set up framework for mock data for java junit tests
REQ-2.2	Setup framework for mock data for Flutter tests
REQ-2.3	Implement unit tests to achieve maximum code coverage on existing code base

3.2.2 Functional Behavior (System-Level)

- The system shall support automated unit testing for existing application components.
- The testing framework shall enable repeatable execution of test suites during development and deployment activities.
- The testing approach shall distinguish between automated tests and user-focused validation activities where applicable.

3.2.3 Constraints and Limitations

- Unit testing shall be limited to existing CareConnect functionality.
- This feature does not introduce new application behavior or modify user-facing functionality.

3.2.4 Acceptance Criteria

The acceptance criteria for the quality assurance and JUnit testing requirement will be considered met when there is a framework implemented for mock data for both the frontend and backend, and code coverage for existing code base is maximized.

Testing Acceptance Criteria (Functional Focus)

AC ID	**Criterion**	**Verification**	**Pass Condition**
AC-3.2.4-001	Java Mock Framework – Backend shall use	Execute unit tests with mocks	All backend modules covered by Java mocks

	Java mocks for testing		
AC-3.2.4-002	Flutter Mock Framework – Frontend shall use Dart/Flutter mocks	Execute frontend unit tests	All frontend modules covered by Dart mocks
AC-3.2.4-003	Backend Coverage – Backend code shall have ≥ 80% test coverage	Coverage reports from CI/CD	Coverage ≥ 80%
AC-3.2.4-004	Frontend Coverage – Frontend code shall have ≥ 70% test coverage	Coverage reports from CI/CD	Coverage ≥ 70%
AC-3.2.4-005	CI/CD Automation – All tests shall run automatically in CI/CD pipeline	Pipeline execution logs	All tests executed successfully
AC-3.2.4-006	Zero Test Skipping – No tests shall be skipped during execution	CI/CD test reports	All tests executed without skipping
AC-3.2.4-007	Documentation – Test documentation shall be complete and up-to-date	Review of test docs	All required test documentation is complete

3.3 Role-Based Access Control (RBAC) Enhancement

3.3.1 Description

CareConnect shall implement a redesigned role-based access control (RBAC) framework to ensure that users can access only those system resources appropriate to their assigned roles. The RBAC enhancement focuses on authorization enforcement rather than the creation of new roles or permissions.

RBAC enforcement applies to both user interface navigation and backend API access to ensure consistent authorization across the system.

CareConnect Currently utilizes four roles admin, patient, family member, and caregiver. Family members have read-only access to the patient's data they are connected to. Patient has control

over their dashboard, the ability to send their caregiver messages, track symptoms, and mark tasks as complete. The caregiver also acts as a type of administrator role over their profile and their patient's profile with the ability to initiate communications and do things like assign tasks. Admin is an internal role and manages logs, compliance and permissions. Please see table below (table 3 from section 2.3 of the SRS from summer/fall 2025) for a description of each role and their responsibilities and expectations.

User Class	**Role Description**	**Core Responsibilities**	**Technical Skill**	**Access**
Caregiver	Manages care, schedules, monitoring, billing	Add patients, create tasks, respond to alerts	Medium-High	Full
Care Receiver	Care recipient	Log symptoms, view schedules, respond to reminders, SOS alerts	Low-Medium	Limited
Family Member	Read-only support role	View reports, monitor logs, receive alerts	Low	Read-only
Admin (Internal)	Backend/system oversight	Manage logs, compliance, permissions	High	Full

The CareConnect system baseline does have roll-based access control implemented. The current system for access control is unorganized and each component for both the frontend and backend are processing access and permissions independently instead of a centralized system where any updates to roll based access control can be handled in a single place. This also does not conform to the object-oriented programming best practice of code reusability.

Requirement ID	Requirement Description
REQ-3	Redesign and implement a new system for RBAC
REQ-3.1	Set up Java class with any methods needed for roll and access processing to better organize RBAC logic on the backend
REQ-3.2	Implement RBAC on each backend endpoint using these class and methods created for REQ-3.1
REQ-3.3	Setup Dart class and methods needed for roll and access processing to organize RBAC for the front end
REQ-3.4	Implement RBAC on each frontend screen using the class and methods created for REQ-3.3

The RBAC implementation shall use the existing role set: Caregiver, Patient, Family Member, and Admin.

RBAC enforcement shall be performed server-side on protected endpoints; client-side hiding of UI elements shall not be treated as sufficient authorization control.

All protected endpoint requests shall be evaluated for authentication and authorization before business logic executes. Requests with missing or invalid authentication tokens shall return 401 (Unauthorized). Requests from authenticated users without sufficient permissions shall return 403 (Forbidden).

3.3.2 RBAC Framework Model

The system shall implement an RBAC framework based on users, roles, and permissions scoped within a tenant. Users are assigned one or more roles, and roles define a collection of permissions that authorize access to application features and backend services.

RBAC configuration is tenant-aware, ensuring that roles, permissions, and access rights are isolated per tenant and enforced consistently throughout the application.

3.3.3 Roles and Permissions

Roles represent logical groupings of permissions that define user access levels within the application. Permissions represent authorization to perform a specific action or access a protected resource, such as viewing a page or invoking an API endpoint.

Application features and backend services shall be protected by permission checks. Users inherit permissions through their assigned roles.

3.3.4 Multi-Tenant RBAC Data Model

The RBAC framework shall support a multi-tenant architecture by logically separating RBAC data per tenant. Users, roles, and permissions shall be associated with a tenant and shall not be shared across tenants.

Logical RBAC entities include:

- Users
- Roles
- Permissions
- User-Role associations
- Role-Permission associations

3.3.5 Functional Requirements

The system shall grant access to application features based on assigned permissions.

The system shall restrict access to application screens based on assigned user roles,

The system shall support assignment of one or more roles per user.

The system shall prevent unauthorized access to backend API endpoints regardless of client-side navigation state,

The system shall redirect users who attempt unauthorized access or provide a clear message with a defined exit path.

The system shall prevent cross-tenant access to RBAC data and protected resources.

The system shall enforce authorization rules consistently across backend components.

Requirement ID	Requirement Description
REQ-3.5	The system shall require valid JWT authentication for all non-public API and WebSocket endpoints. Public authentication endpoints shall be limited to registration and password recovery/reset flows.
REQ-3.6	The system shall enforce role and relationship scope (self, managed_patient, assigned_patient, admin) on user profile and permissions endpoints.
REQ-3.7	The system shall enforce role-restricted access for dashboard and module endpoints (caregiver, patient, and admin access as applicable).

REQ-3.8	The system shall restrict all /admin/* endpoints to the Admin role only.
REQ-3.9	The system shall return 401 Unauthorized for missing/invalid/expired authentication and 403 Forbidden for insufficient role, permission, or relationship scope.
REQ-3.10	The system shall enforce UI route guards that mirror backend RBAC rules and restrict access to application screens based on assigned user roles.
REQ-3.11	The system shall redirect users who attempt unauthorized access or provide a clear access-denied message with a defined exit path.

Protected Endpoint Families:

Family	Endpoint Pattern(s)	Access Rule
Public Auth	/auth/register, /auth/password-recovery, /auth/password-reset	Public
Auth Session Flows	/auth/login, /auth/refresh, /auth/logout	Authentication flow endpoints; enforce token requirements per endpoint behavior
User & Permissions	/users/{userId}, /users/{userId}/permissions	Self or scoped relationship; admin unrestricted
Dashboard/Role Modules	/dashboard/*, subscriptions/billing, analytics	Role-restricted by module
Patient-Scoped Clinical Data	/tasks*, /health*, /ai/insights/{patientId}, /health/reports/{patientId}	Role + relationship scope required
Admin	/admin/*	Admin only
Real-Time & Notifications	wss://.../ws, /api/v1/notifications/send*	JWT required; sender/recipient scope enforced

Protected Pages

Page:	**Role Required:**

Patient Dashboard	patient
Caregiver Dashboard	caregiver
Admin Console	caregiver
Patient Profile Management	caregiver

3.3.6 Acceptance Criteria

The acceptance criteria for the RBAC model requirement will be considered met when the business logic for processing rolls and determining whether to grant access or not is organized into a Java class and a dart class for the frontend and backend respectively, and the methods in these classes are implemented throughout the application for access control purposes. Additionally, there should be no difference from a user perspective meaning all features are working properly and there should be no degradation in performance.

In addition to the criteria above, RBAC acceptance shall also require consistent authorization outcomes across protected APIs: missing, invalid, or expired authentication must return `401 Unauthorized`, and valid authentication with insufficient role, permission, or relationship scope must return `403 Forbidden`. This behavior shall be applied consistently for profile, dashboard, and admin-protected operations.

RBAC acceptance shall further require equivalent enforcement in the UI layer so unauthorized route access is blocked. When unauthorized access is attempted, the system shall either redirect the user or display a clear access-denied message with a defined exit path. Completion evidence shall include both positive and negative RBAC test scenarios for API and UI behavior.

3.4 Scope Integrity Statement

The system features defined in this section represent non-functional enhancements that enable and protect existing CareConnect functionality. These features do not alter the system's business requirements, user workflows, or functional outputs as defined in prior SRS versions.

Any functionality not explicitly described in this section or in prior CareConnect SRS documents is outside the scope of the Spring 2026 v1 release.

3.5 Scheduling & Notifications

The CareConnect system shall provide functionality for creating, managing, and tracking scheduled care-related tasks. Caregivers shall be able to create both predefined template-based

tasks (e.g., medications, meals, exercises) and custom tasks tailored to individual patient needs. Tasks may be scheduled as one-time, recurring, or conditional events.

The system shall deliver reminders and alerts to patients and caregivers using supported notification channels, including push notifications, email, and SMS, based on user preferences and task priority. Patients shall be able to mark tasks as completed, missed, or deferred, and caregivers shall be notified of missed or overdue tasks.

Future enhancements may include caregiver shift coordination features to support professional caregiving teams, enabling handoff awareness and continuity of care.

3.6 Health Data Tracking

CareConnect shall allow patients and caregivers to record and monitor health-related data, including symptoms, vital signs, meals, and mood indicators. Data entry may be performed manually or automatically through supported device integrations.

The system shall store health data securely and present it in an organized, time-based format that supports trend analysis. Caregivers shall be able to review historical records and identify changes or anomalies in patient health patterns.

The system shall support configurable alerts that notify caregivers when logged data exceeds predefined thresholds or indicates potential risk, such as negative mood trends or abnormal symptom patterns.

3.7 AI Integration

CareConnect shall include AI-powered features to assist patients and caregivers with health-related information and insights. AI capabilities may include mood detection during video interactions and an “Ask AI” feature that responds to user questions using available patient data.

AI-generated responses shall be informational only and shall not provide clinical diagnoses or treatment recommendations. The system shall include appropriate disclaimers and escalation mechanisms when AI-generated content falls outside predefined safety boundaries.

High-risk or ambiguous AI interactions may be flagged for human review to ensure patient safety and compliance with healthcare regulations.

3.8 Communication & Media

The system shall support secure communication between caregivers, patients, and authorized family members. Communication features shall include in-app messaging, voice calls, and video calls.

CareConnect shall allow users to share media such as images, documents, and notes related to patient care. All communication and media exchanges shall be encrypted and access-controlled based on user roles.

The system shall include emergency communication functionality, allowing patients to trigger an SOS alert that notifies caregivers and designated contacts. Future enhancements may include telehealth integration with third-party platforms for virtual medical consultations.

3.9 Device & Third-Party Integrations

CareConnect shall integrate with external devices and third-party systems to enhance health monitoring and care coordination. Supported integrations may include wearable devices for activity and vital tracking, home monitoring systems for safety and environmental awareness, and medication reference systems.

The system shall retrieve data from integrated services using secure APIs and user-authorized permissions. Users shall be able to manage and revoke device or service connections through system settings.

Future integrations may include smart home devices, pharmacy systems, and additional healthcare platforms as supported by available APIs.

3.10 Gamification

CareConnect shall incorporate gamification elements to encourage patient engagement and adherence to care plans. Gamification features may include points, badges, achievement milestones, and motivational messages.

The system shall reward consistent task completion and healthy behaviors while avoiding competitive pressure that could negatively impact users. Caregivers may view engagement summaries to better understand patient participation and motivation levels.

Gamification features shall be optional and configurable to accommodate different user preferences and care contexts.

3.11 Analytics & Reporting

The system shall generate analytics and reports to support caregivers in monitoring patient progress and system usage. Dashboards shall display key metrics such as task completion rates, health trends, and engagement levels.

CareConnect shall allow authorized users to export reports in standard formats such as CSV or PDF for external review or consultation. Reports may be generated on demand or through scheduled processes.

The system shall support both real-time data processing for immediate insights and batch processing for historical analysis and long-term reporting.

3.12 Infrastructure, Security & Compliance

CareConnect shall be deployed using secure, scalable cloud infrastructure designed to support high availability and reliability. All sensitive data shall be encrypted both at rest and in transit using industry-standard cryptographic methods.

The system shall enforce compliance with applicable regulations, including HIPAA and GDPR, through role-based access control, audit logging, and data minimization practices. Backup and disaster recovery mechanisms shall be implemented to protect against data loss and service interruptions.

The platform shall include monitoring and logging capabilities to support operational visibility, incident response, and compliance audits.

3.13 Social Networking

CareConnect shall provide limited social networking functionality focused on care coordination and emotional support. Features may include secure group communication, shared calendars, and care-related activity feeds.

The system shall support emotional check-ins that allow patients to express their well-being and caregivers to respond appropriately. All social interactions shall remain private, access-controlled, and limited to authorized users.

Public social media integration and open community forums are outside the scope of the system.

---

4. External Interface Requirements

4.1 User Interfaces Overview

CareConnect shall provide a set of primary user interfaces that support the core workflows for patients, caregivers, and authorized family members. The major screens shall include onboarding and authentication screens (welcome, registration, login, and password recovery), role-specific dashboards, scheduling and task management views, health tracking interfaces, messaging and communication screens, and account settings.

Navigation shall be consistent across the mobile and web applications, using a clear menu structure that allows users to move between dashboards, tasks, messages, health data, and profile settings with minimal effort. Screen layouts shall be responsive and adapt to different device sizes while maintaining visual consistency (as cited in the CareConnect SRS, Aug 30, 2025).

4.2 Hardware Interfaces

CareConnect shall support integration with external physical devices, including wearable health trackers and compatible home monitoring devices where applicable. Wearable devices such as Fitbit shall communicate health data (e.g., heart rate, activity, steps, SpO_2) to the system through secure APIs.

The system shall be capable of receiving and processing data from connected devices, storing relevant metrics, and using them for notifications and analytics. Device connectivity shall be configurable by the user, and permissions shall be required before collecting or sharing any device data (as cited in the CareConnect SRS, Aug 30, 2025).

4.3 Software Interfaces

CareConnect shall integrate with external software systems and services through well-defined APIs. Key third-party integrations shall include:

- Stripe for billing and subscription management.
- AWS services for authentication, storage, and backend processing.
- Google Calendar for shared scheduling and task coordination.
- USPS Informed Delivery (where applicable) for accessibility-related mail assistance.

All integrations shall use secure, authenticated communication methods and follow industry-standard API practices. The system shall be designed to allow future expansion to additional third-party services without major architectural changes (as cited in the CareConnect SRS, Aug 30, 2025).

4.4 Communications Interface

CareConnect shall use secure, standardized communication protocols for all data exchange between clients, servers, and third-party services. All network communications shall be encrypted using TLS 1.3 or an equivalent secure protocol.

Data exchanged between components shall use structured formats such as JSON over REST-based APIs. Real-time features, such as notifications or live updates, shall use appropriate messaging mechanisms such as push notifications or event-driven communication where applicable (as cited in the CareConnect SRS, Aug 30, 2025).

4.5 Operations

CareConnect shall support continuous operation through automated monitoring, logging, and maintenance processes. The system shall generate logs for security events, system errors, user activity, and performance metrics to support troubleshooting and compliance (as cited in the CareConnect SRS, Aug 30, 2025).

Cloud-based monitoring tools shall track system health, resource usage, and potential failures in real time. The platform shall support automated deployments, backups, and system updates with minimal manual intervention (as cited in the CareConnect SRS, Aug 30, 2025).

4.6 Reporting Requirements

CareConnect shall provide system-generated reports and analytics outputs for caregivers and authorized users. The system shall generate real-time dashboard metrics related to task adherence, symptom trends, and patient activity.

Users shall be able to export reports in standard formats such as CSV and PDF for external use or consultation. Reports shall be generated on demand or through scheduled processes as configured by authorized users (as cited in the CareConnect SRS, Aug 30, 2025).

4.7 Site Adaptation

CareConnect shall be configurable to operate in different environments, including development, testing, and production deployments. The system shall support regional variations such as data residency requirements, time zones, and language preferences where applicable.

Configuration settings shall be managed through secure administrative controls rather than hard-coded values, allowing for flexibility across different deployments and future expansion (as cited in the CareConnect SRS, Aug 30, 2025).

4.8 Business Rules

CareConnect shall enforce business rules governing user access, data handling, and system workflows. Access to features and data shall be controlled through role-based permissions, ensuring that patients, caregivers, and family members can only view or modify authorized information.

Billing rules shall define when subscriptions are activated, how payments are processed, and how failures are handled. Data handling rules shall ensure compliance with privacy and security requirements, including data retention and user consent.

Workflow rules shall govern how tasks, notifications, and alerts are created, assigned, and resolved within the system (as cited in the CareConnect SRS, Aug 30, 2025).

---

5. System Features / Modules

This section contains **detailed functional specifications** for each module.

Each module includes:

- Description
- Stimulus/Response Sequences
- Functional Requirements
- Sequence Diagrams (where applicable)

5.1 Onboarding & Authentication

Description

Onboarding and authentication ensure that caregivers, patients, and family members can securely access their roles within CareConnect. The system supports email/password registration, login, and password recovery.

Stimulus/Response Sequences

- **Stimulus:** User provides email and password for registration/login.
- **Response:** System authenticates credentials and redirects to the appropriate dashboard.
- **Stimulus:** User clicks "Forgot Password."
- **Response:** System prompts for email, sends a reset link, and confirms password reset.

Functional Requirements

- Email and password registration and login
- Password recovery process with email confirmation
- Session management to prevent unauthorized access
- Multi-factor authentication (MFA) for enhanced security (future enhancement)

(as cited in the CareConnect SRS, Aug 30, 2025)

5.2 Billing & Subscription Management

Description

This module handles subscription management for caregivers, including pricing tiers, billing cycle management, and payment collection through secure channels like Stripe.

Stimulus/Response Sequences

- **Stimulus:** Caregiver selects a subscription tier.
- **Response:** System processes payment via Stripe and activates the subscription.

Functional Requirements

- Subscription options: monthly, per-patient
- Integration with Stripe for payment collection
- Subscription activation upon user onboarding
- Automatic billing notifications and reminders
- Billing history view for users

(as cited in the CareConnect SRS, Aug 30, 2025)

5.3 User & Role Management

Description

This module ensures that users are assigned specific roles and permissions based on their responsibilities within CareConnect (e.g., caregiver, patient, family member).

Stimulus/Response Sequences

- **Stimulus:** User is assigned a role during registration.
- **Response:** System assigns permissions based on role and restricts access to unauthorized features.

Functional Requirements

- Role-based access control (RBAC)
- Role management by administrators
- Permission enforcement across modules (e.g., patients cannot access caregiver settings)
- User activity logs to support auditing and compliance

(as cited in the CareConnect SRS, Aug 30, 2025)

5.4 Dashboards

Description

Caregivers and patients have personalized dashboards to view important information, track progress, and manage daily tasks.

Stimulus/Response Sequences

- **Stimulus:** Caregiver logs in and accesses the dashboard.
- **Response:** System displays a dashboard with tasks, health metrics, and notifications.

Functional Requirements

- Personalized dashboards for caregivers and patients
- Task overview, health tracking, and calendar view
- Real-time updates for critical alerts or health metrics
- Ability to update patient records and assign tasks from the dashboard

(as cited in the CareConnect SRS, Aug 30, 2025)

5.5 Scheduling & Notifications

Description

This module supports scheduling of tasks and sends reminders via various channels (push notifications, emails, SMS).

Stimulus/Response Sequences

- **Stimulus:** Caregiver creates a scheduled task for the patient.
- **Response:** Patient receives a notification and reminder about the task at the scheduled time.

Functional Requirements

- Task scheduling with flexible timing and recurrence options
- Customizable reminder settings (push, email, SMS)
- Task completion notifications
- Missed task alerts for caregivers

(as cited in the CareConnect SRS, Aug 30, 2025)

5.6 Health Data Tracking

Description

Allows caregivers and patients to track and log health data (e.g., vital signs, symptoms) and generate reports.

Stimulus/Response Sequences

- **Stimulus:** Caregiver enters patient health data (e.g., blood pressure).
- **Response:** System logs the data and displays it in a graph for easy visualization.

Functional Requirements

- Data entry for health metrics (e.g., weight, blood pressure, symptoms)
- Real-time tracking and visualization of health data
- Reporting and export options (CSV, PDF)
- Integration with wearables and external devices for automated tracking (future enhancement)

(as cited in the CareConnect SRS, Aug 30, 2025)

5.7 AI Integration

Description

This module integrates AI features like mood detection, predictive analytics, and query handling to assist caregivers and patients.

Stimulus/Response Sequences

- **Stimulus:** Caregiver asks AI about patient symptoms.
- **Response:** AI returns insights based on patient data and symptoms.

Functional Requirements

- AI-based symptom analysis and health insights
- Mood detection during video calls using facial recognition

- "Ask AI" feature for health-related queries
- Trigger word detection for automatic task generation

(as cited in the CareConnect SRS, Aug 30, 2025)

5.8 Communication & Media

Description

This module supports communication features like in-app messaging, video calls, and media sharing.

Stimulus/Response Sequences

- **Stimulus:** Caregiver sends a message to the patient.
- **Response:** Patient receives the message in the app and is notified.

Functional Requirements

- In-app messaging for real-time communication
- Video and audio calling for telemedicine
- Media upload (e.g., images, documents)
- Encryption of communications to ensure privacy and compliance

(as cited in the CareConnect SRS, Aug 30, 2025)

5.9 Device & Third-Party Integrations

Description

Integrates external devices (wearables, home monitoring) and third-party services (e.g., Stripe for payments).

Stimulus/Response Sequences

- **Stimulus:** Wearable data is synced with the app.
- **Response:** System displays health data from the wearable on the dashboard.

Functional Requirements

- Integration with wearables for automatic health tracking (e.g., Fitbit, Apple HealthKit)
- Integration with third-party APIs like Stripe for payments, Nest for smart home control
- Secure authentication and data sharing

(as cited in the CareConnect SRS, Aug 30, 2025)

5.10 Gamification

Description

The gamification module enhances user engagement through rewards, badges, and leaderboards.

Stimulus/Response Sequences

- **Stimulus:** User completes a task.
- **Response:** System awards points, a badge, or updates the leaderboard.

Functional Requirements

- Rewards for completing tasks or achieving health goals
- Badges and motivational messages
- Leaderboards for patient and caregiver engagement
- Optional challenges and social sharing of achievements

(as cited in the CareConnect SRS, Aug 30, 2025)

5.11 Social Networking

Description

This module allows for private messaging and activity feeds, enabling communication between caregivers, patients, and family members.

Stimulus/Response Sequences

- **Stimulus:** User posts a care update in the activity feed.

- **Response:** Feed is updated for all relevant users.

Functional Requirements

- Private messaging for secure caregiver-patient communication
- Basic activity feed for care updates
- Support for group discussions (caregiver communities, support groups)

(as cited in the CareConnect SRS, Aug 30, 2025)

5.12 Analytics & Reporting

Description

This module generates analytics and reports, providing valuable insights for caregivers and healthcare professionals.

Stimulus/Response Sequences

- **Stimulus:** Caregiver requests a report on patient health data.
- **Response:** System generates a report and provides download/export options.

Functional Requirements

- Real-time analytics for health and care metrics
- Data export (CSV, PDF) for external consultation
- Dashboard with visualizations of patient health trends
- Periodic reports for caregivers and family members

(as cited in the CareConnect SRS, Aug 30, 2025)

6. Nonfunctional Requirements (Inherited from Summer/Fall 2025)

6.1 Application Hosting and Deployment Modernization

6.1.1 Description

CareConnect shall support deployment using a managed AWS hosting service selected from AWS Fargate or AWS App Runner. This system feature governs the operational environment of the application and does not alter existing business logic or user workflows.

The purpose of this feature is to ensure that CareConnect operates within acceptable operational, maintenance, and cost constraints while supporting scalable and reliable deployment.

6.1.2 Functional Behavior (System-Level)

- **NFR-1.1 Deployment Success Rate**
  Requirement: The system shall support deployment of the existing CareConnect application to the selected AWS hosting service with **≥99% success rate** across deployments.
  Measurement: (Successful Deployments / Total Deployment Attempts) × 100
  Target: 99% or higher

NFR-1.2 Zero-Downtime Migration

Requirement: The system shall migrate to the new hosting service with ≤5 minutes of planned downtime during off-peak hours.
Measurement: Total system unavailability during migration
Target: ≤5 minutes (preferably zero downtime using blue-green deployment)

NFR-1.3 Post-Migration Availability

Requirement: The system shall remain accessible to users following migration with **≥99.5% uptime** measured monthly.
Measurement: (Total Time - Downtime) / Total Time × 100
Target: 99.5% or higher per month

NFR-1.4 Deployment Time

Requirement: The system shall support repeatable deployment processes completing within ≤15 minutes from code commit to production availability.
Measurement: Time from CI/CD pipeline trigger to health check success
Target: ≤15 minutes for standard deployments

NFR-1.5 Rollback Capability

Requirement: The system shall support rollback to previous version within ≤5 minutes in case of deployment failure.
Measurement: Time from rollback initiation to previous version operational
Target: ≤5 minutes

6.1.3 Constraints and Limitations

- This feature does not include redesigning the application architecture or business logic.
- Full application containerization, database re-platforming, or infrastructure redesign beyond the selected hosting service is excluded.
- Deployment changes shall not introduce new application features or modify existing functionality.

6.2 Quality Assurance and JUnit Testing

6.2.1 Description

CareConnect shall incorporate a redesigned Junit testing paradigm focused on improving test structure, coverage, and maintainability for existing system functionality. This feature addresses how the system is validated rather than how it behaves from a user perspective.

The objective of this feature is to provide a consistent and repeatable testing approach that supports regression detection and long-term maintainability.

6.2.2 Functional Behavior (System-Level)

- **NFR-2.1 Unit Test Coverage**
  Requirement: The system shall maintain ≥80% code coverage for backend services measured by line coverage.
  Measurement: (Lines Covered by Tests / Total Lines of Code) × 100
  Target: 80% minimum, 85% goal

NFR-2.2 Frontend Test Coverage
Requirement: The system shall maintain ≥70% code coverage for frontend components.
Measurement: (Components/Functions Covered / Total Components/Functions) × 100
Target: 70% minimum, 75% goal

NFR-2.3 Test Execution Time
Requirement: The complete unit test suite shall execute in ≤5 minutes for rapid feedback.
Measurement: Total execution time for all unit tests
Target: ≤5 minutes

NFR-2.4 Test Success Rate
Requirement: Unit tests shall maintain ≥95% pass rate in the main branch.
Measurement: (Passed Tests / Total Tests) × 100
Target: 95% or higher

NFR-2.5 Test Reliability
Requirement: Flaky tests (intermittent failures) shall represent ≤2% of total test suite.
Measurement: (Flaky Tests / Total Tests) × 100
Target: ≤2%

NFR-2.6 Critical Path Coverage
Requirement: All critical user workflows shall have 100% integration test coverage.
Measurement: (Critical Paths with Tests / Total Critical Paths) × 100
Target: 100%

6.3 Data Encryption

CareConnect shall encrypt all sensitive and health-related data both at rest and in transit. Data in transit shall use TLS 1.3 or an equivalent secure protocol, and data at rest shall be protected using AES-256 or a comparable standard with secure key management.

Access to encrypted data shall be role-based and logged for security monitoring and compliance purposes (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-3.1 Encryption at Rest

Requirement: CareConnect shall encrypt 100% of sensitive and health-related data at rest using AES-256 or stronger.
Measurement: (Encrypted Data Stores / Total Data Stores) × 100
Target: 100%
Standard: AES-256 with AWS KMS key management

NFR-3.2 Encryption in Transit

Requirement: CareConnect shall encrypt 100% of data transmissions using TLS 1.2 or higher.

Measurement: (Encrypted Connections / Total Connections) × 100

Target: 100%

Standard: TLS 1.2 minimum, TLS 1.3 preferred

NFR-3.3 Key Rotation

Requirement: Encryption keys shall be rotated automatically every 90 days.

Measurement: Days since last key rotation

Target: ≤90 days

NFR-3.4 Access Logging

Requirement: Access to encrypted data shall be logged with 100% coverage for audit trail.

Measurement: (Logged Access Events / Total Access Events) × 100

Target: 100%

NFR-3.5 Encryption Performance Impact

Requirement: Encryption/decryption operations shall add ≤50ms latency to data operations.

Measurement: Response time with encryption - Response time without encryption

Target: ≤50ms overhead

6.4 Regulatory Compliance

CareConnect shall comply with HIPAA, GDPR, and other applicable data protection regulations. The system shall use HIPAA-eligible services, role-based access control, secure authentication, and comprehensive audit logging.

For GDPR, the system shall support user rights such as data access, correction, and deletion where applicable, and limit data collection to what is necessary (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-4.1 HIPAA Compliance

Requirement: CareConnect shall maintain 100% compliance with HIPAA Security Rule requirements.

Measurement: (Compliant Controls / Required Controls) × 100

Target: 100%

Audit Frequency: Annual third-party audit

NFR-4.2 GDPR Compliance

Requirement: CareConnect shall respond to GDPR data subject requests within 30 days of receipt.

Measurement: Average response time for data access/deletion requests

Target: ≤30 days (legal requirement)

NFR-4.3 Audit Log Retention

Requirement: The system shall retain audit logs for a minimum of 7 years for HIPAA compliance.

Measurement: Age of oldest retained audit log

Target: ≥7 years

NFR-4.4 Access Control Audit

Requirement: Role-based access controls shall be audited quarterly with 100% role verification**.**

Measurement: (Verified Roles / Total Roles) × 100 per quarter

Target: 100% quarterly

NFR-4.5 Data Breach Notification

Requirement: The system shall detect and report potential data breaches within **72 hours** (GDPR requirement).

Measurement: Time from breach detection to notification

Target: ≤72 hours

NFR-4.6 Business Associate Agreements

Requirement: 100% of third-party services handling PHI shall have executed BAAs.

Measurement: (Services with BAAs / Total Services Handling PHI) × 100

Target: 100%

6.5 Accessibility

CareConnect shall follow basic accessibility best practices to support usability for a broad range of users. The interface shall prioritize clear navigation, readable text, and sufficient color contrast.

Full WCAG 2.1 Level AA compliance is not required for the MVP and is deferred to future releases (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-5.1 Color Contrast

Requirement: Text elements shall maintain minimum 4.5:1 contrast ratio for normal text and 3:1 for large text.
Measurement: WCAG contrast ratio calculation
Target: 4.5:1 normal text, 3:1 large text (WCAG 2.1 Level AA)

NFR-5.2 Keyboard Navigation

Requirement: 100% of interactive elements shall be accessible via keyboard navigation.
Measurement: (Keyboard-Accessible Elements / Total Interactive Elements) × 100
Target: 100%

NFR-5.3 Screen Reader Compatibility

Requirement: The system shall be compatible with JAWS, NVDA, and VoiceOver screen readers with ≥90% content accessibility.
Measurement: Screen reader testing coverage
Target: 90% of content accessible via screen readers

NFR-5.4 Text Scaling

Requirement: The interface shall remain functional when text is scaled up to 200% without loss of content or functionality.
Measurement: Visual regression testing at 200% zoom
Target: Full functionality maintained at 200% zoom

NFR-5.5 Touch Target Size

Requirement: Interactive elements shall be minimum 44×44 pixels for mobile touch targets.
Measurement: Pixel dimensions of touch targets
Target: ≥44×44 pixels (WCAG 2.1 Level AAA guideline)

6.6 Offline Mode

CareConnect shall provide limited offline functionality when network connectivity is unavailable. Users shall be able to view cached schedules, notes, and basic patient information while offline.

Any data entered offline shall be synchronized automatically once connectivity is restored, with clear user notifications about synchronization status (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-6.1 Offline Data Availability

Requirement: Users shall access **≥**80% of frequently used features in offline mode.
Measurement: (Available Features Offline / Total Features) × 100
Target: 80% or higher

NFR-6.2 Cached Data Freshness

Requirement: Cached offline data shall be **≤**24 hours old when network connectivity is available.
Measurement: Time since last data synchronization
Target: ≤24 hours

NFR-6.3 Synchronization Time

Requirement: Offline data shall synchronize within **≤**30 seconds after connectivity restoration for typical usage.
Measurement: Time from connectivity restored to sync complete
Target: ≤30 seconds for <100 records

NFR-6.4 Conflict Resolution

Requirement: Offline-online data conflicts shall be detected with 100% accuracy and presented to user for resolution.
Measurement: (Detected Conflicts / Actual Conflicts) × 100
Target: 100%

NFR-6.5 Offline Storage Limit

Requirement: Offline cache shall not exceed 500 MB per user on mobile devices.
Measurement: Total offline cache size
Target: ≤500 MB

NFR-6.6 Sync Status Visibility

Requirement: Synchronization status shall be visible to user within **≤**2 seconds of status change.
Measurement: Time from status change to UI update
Target: ≤2 seconds

6.7 UI/UX Notes

The user interface shall be intuitive, consistent, and easy to navigate across all screens and platforms. The system shall minimize complexity and clearly highlight important actions such as task completion and emergency alerts.

Users shall receive clear feedback through loading indicators, success messages, and error notifications (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-7.1 User Task Completion

Requirement: ≥85% of users shall complete primary tasks without assistance within 3 attempts.
Measurement: User testing success rate
Target: 85% task completion rate

NFR-7.2 Navigation Depth

Requirement: Critical features shall be accessible within **≤**3 clicks/taps from the home screen.
Measurement: Click/tap count from home to feature
Target: ≤3 clicks/taps

NFR-7.3 Loading Indicators

Requirement: Loading indicators shall appear within **≤**200m**s** for operations taking **>**1 second.
Measurement: Time from operation start to indicator display
Target: ≤200ms

NFR-7.4 Error Message Clarity

Requirement: **≥**90% of error messages shall include actionable guidance for resolution.
Measurement: (Error Messages with Guidance / Total Error Messages) × 100
Target: 90% or higher

NFR-7.5 Interface Consistency

Requirement: **≥**95% of UI elements shall follow established design system patterns.
Measurement: (Compliant UI Elements / Total UI Elements) × 100
Target: 95% or higher

NFR-7.6 Mobile Responsiveness

Requirement: The interface shall render correctly on 100% of supported device sizes (320px to 2560px width).
Measurement: Visual regression testing across viewports
Target: 100% correct rendering

6.8 Backup & Disaster Recovery

CareConnect shall implement regular automated backups of critical system data, stored securely in geographically separate locations.

The system shall define Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO), and conduct periodic disaster recovery testing (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-8.1 Backup Frequency

Requirement: Critical data shall be backed up every 24 hours with incremental backups **every 6 hours**.
Measurement: Time between backups
Target: ≤24 hours full, ≤6 hours incremental

NFR-8.2 Backup Retention

Requirement: Daily backups shall be retained for 30 days, weekly backups for 1 year.
Measurement: Age of oldest available backup
Target: 30 days daily, 365 days weekly

NFR-8.3 Recovery Time Objective (RTO)

Requirement: The system shall recover from major failure within **≤**4 hours of incident declaration.
Measurement: Time from failure to full system restoration
Target: ≤4 hours

NFR-8.4 Recovery Point Objective (RPO)

Requirement: Data loss shall not exceed **≤**1 hour of transactions in disaster scenarios.
Measurement: Age of most recent restorable backup
Target: ≤1 hour data loss maximum

NFR-8.5 Backup Verification

Requirement: 100% of backups shall be tested for recoverability quarterly.

Measurement: (Tested Backups / Total Backups) × 100 per quarter
Target: 100% tested quarterly

NFR-8.6 Geographic Redundancy

Requirement: Backups shall be stored in ≥2 geographically separate AWS regions (minimum 250 miles apart).
Measurement: Number of backup regions and distance between them
Target: ≥2 regions, ≥250 miles separation

NFR-8.7 DR Testing

Requirement: Disaster recovery procedures shall be tested semi-annually with ≥90% success rate.
Measurement: (Successful DR Tests / Total DR Tests) × 100
Target: 90% or higher, tested every 6 months

6.9 Performance & Scalability Targets

CareConnect shall deliver responsive performance under normal usage and support multiple concurrent users without significant delays.

The system shall use scalable cloud infrastructure to handle varying demand efficiently and support future growth without major redesign (as cited in the CareConnect SRS, Aug 30, 2025).

NFR-9.1 Response Time - API Endpoints

Requirement: 95% of API requests shall respond within ≤500ms under normal load.
Measurement: 95th percentile API response time
Target: ≤500ms (P95)

NFR-9.2 Response Time - Page Load

Requirement: 90% of page loads shall complete within ≤2 seconds on 4G connections.
Measurement: 90th percentile page load time
Target: ≤2 seconds (P90)

NFR-9.3 Response Time - Database Queries

Requirement: 95% of database queries shall execute within ≤100ms.
Measurement: 95th percentile query execution time
Target: ≤100ms (P95)

NFR-9.4 Concurrent Users

Requirement: The system shall support ≥1,000 concurrent users without performance degradation.
Measurement: Load testing with concurrent user simulation
Target: 1,000+ concurrent users with response times within SLA

NFR-9.5 Throughput

Requirement: The system shall handle ≥100 transactions per second (TPS) during peak hours.
Measurement: Transactions processed per second
Target: ≥100 TPS sustained

NFR-9.6 Scalability - Horizontal

Requirement: The system shall auto-scale from 2 to 20 instances based on CPU utilization (>70% triggers scale-up).
Measurement: Instance count, CPU utilization, scale-up/down events
Target: Auto-scale within 5 minutes of threshold breach

NFR-9.7 Resource Utilization

Requirement: Average CPU utilization shall remain ≤60% under normal load to allow headroom for spikes.
Measurement: Average CPU utilization over 24-hour period
Target: ≤60% average

NFR-9.8 Memory Utilization

Requirement: Average memory utilization shall remain ≤70% under normal load.
Measurement: Average memory utilization over 24-hour period
Target: ≤70% average

NFR-9.9 Database Connections

Requirement: Database connection pool shall maintain ≤80% utilization under peak load.
Measurement: (Active Connections / Max Pool Size) × 100
Target: ≤80%

NFR-9.10 Error Rate

Requirement: System error rate shall be ≤0.1% of total requests under normal operation.
Measurement: (Failed Requests / Total Requests) × 100

Target: ≤0.1%

NFR-9.11 Time to First Byte (TTFB)

Requirement: 90% of requests shall receive first byte within **≤**200ms.

Measurement: 90th percentile TTFB

Target: ≤200ms (P90)

NFR-9.12 CDN Cache Hit Rate

Requirement: Static assets shall achieve **≥**90% CDN cache hit rate.

Measurement: (Cache Hits / Total Requests) × 100

Target: ≥90%

7. Future Scenario / Functional Requirements (Inherited from Summer/Fall 2025)

This section describes features and capabilities that are planned for future releases of CareConnect beyond the initial Minimum Viable Product (MVP). These enhancements are intended to expand system functionality, improve usability, and increase integration with external technologies as resources and time allow.

7.1 Multilingual Support

Future versions of CareConnect shall support multiple languages beyond English. Users shall be able to select their preferred language from within application settings, and all user interface text, notifications, and system messages shall be accurately translated. The system architecture shall be designed to accommodate additional languages without requiring major redesign (as cited in the CareConnect SRS, Aug 30, 2025).

7.2 Health Simulator Integration

CareConnect may integrate with external health simulation platforms in future releases to support predictive modeling, training scenarios, and simulated patient outcomes. This feature could assist caregivers in decision-making, risk assessment, and care planning (as cited in the CareConnect SRS, Aug 30, 2025).

7.3 Home Monitoring Integration

Future iterations of CareConnect shall support deeper integration with home monitoring systems, including cameras, motion sensors, and environmental monitoring devices. Relevant safety and activity data from these systems shall be securely transmitted to and displayed within CareConnect (as cited in the CareConnect SRS, Aug 30, 2025).

7.4 Smart Home Integration

CareConnect may integrate with smart home ecosystems such as Nest, Ring, or similar platforms in future releases. This could enable automated alerts for events such as falls, unusual activity, or environmental hazards, providing caregivers with enhanced situational awareness (as cited in the CareConnect SRS, Aug 30, 2025).

7.5 Wearables & Health Metrics Enhancement

While basic wearable integration may exist in the MVP, future enhancements shall allow for more advanced analysis of health metrics from devices such as smartwatches or fitness trackers. This may include trend analysis, predictive insights, and richer visualizations of patient health data (as cited in the CareConnect SRS, Aug 30, 2025).

7.6 Patient Linking

Future versions of CareConnect shall improve patient-caregiver linking by introducing additional secure methods such as one-time codes, automated matching systems, or integration with healthcare institutions. The goal is to streamline onboarding and relationship establishment (as cited in the CareConnect SRS, Aug 30, 2025).

7.7 Reminder & Alert System

The reminder and alert system shall be enhanced to include more intelligent scheduling, customizable thresholds, and adaptive notification logic based on user behavior. These improvements aim to reduce alert fatigue while ensuring critical reminders are delivered effectively (as cited in the CareConnect SRS, Aug 30, 2025).

7.8 Virtual Check-In Rounds

CareConnect may expand virtual check-in capabilities to support structured, recurring wellness assessments conducted remotely by caregivers. These check-ins could include standardized questionnaires, video check-ins, or automated monitoring workflows (as cited in the CareConnect SRS, Aug 30, 2025).

7.9 Voice-Activated Commands

Future versions of CareConnect may include voice-activated functionality, allowing users to interact with the system through spoken commands. This could support hands-free creation of tasks, symptom logging, messaging, and information retrieval (as cited in the CareConnect SRS, Aug 30, 2025).

7.10 Telehealth Bridge

CareConnect shall be designed to integrate with external telehealth platforms such as Zoom or Microsoft Teams in future releases. This may include features such as call transcription, automated note-taking, and secure data sharing with healthcare providers (as cited in the CareConnect SRS, Aug 30, 2025).

7.11 Caregiver Shift Scheduling

Future enhancements may introduce a dedicated caregiver shift scheduling module to support professional caregiving teams. This would allow caregivers to coordinate schedules, manage handoffs, and ensure continuous patient coverage (as cited in the CareConnect SRS, Aug 30, 2025).

7.12 Meal & Nutrition Tracking

CareConnect may expand meal and nutrition tracking by allowing users to log detailed dietary information, track nutritional intake, and receive insights based on recorded data. This could support improved health management and dietary planning (as cited in the CareConnect SRS, Aug 30, 2025).

7.13 AI Mood Detection

Future versions of CareConnect may incorporate AI-based mood detection using facial expression analysis, voice tone recognition, or behavioral patterns. This feature could help caregivers identify potential emotional or mental health concerns (as cited in the CareConnect SRS, Aug 30, 2025).

7.14 Gamification

Gamification features may be further developed to include additional reward systems, challenges, and social elements. This could involve personalized achievement pathways, competitive elements, and enhanced motivational tools for patients and caregivers (as cited in the CareConnect SRS, Aug 30, 2025).

7.15 Medication Management

Future releases of CareConnect may include a comprehensive medication management module with features such as medication tracking, automated refill reminders, interaction warnings, and pharmacy system integration to improve adherence and safety (as cited in the CareConnect SRS, Aug 30, 2025).

Software Requirements Specification

for

Care Connect

Version 1.0

University Of Maryland Global Campus

SWEN 670 – Software Engineering Capstone

January 24, 2026

Contributors:
Justinah Bashua (Technical Lead/Architect)
Corey Bayliss (Lead Tester)
Matthew Lester (Lead Software Developer)
James Stevens (Team Lead)
Brandon Sutan (Lead Business Analyst)

Revision History

Date	Version	Reason for Change	Team/Author
1/24/2026	1.0	Initial document	All team members
2/4/2026	1.1	Incorporated feedback	James Stevens

Table of Contents

1. Introduction

This document provides a formal specification of requirements for a defined set of enhancements to the CareConnect platform. It establishes the scope, context, and constraints under which the enhancements are to be designed, implemented, and validated, and serves as a foundation for technical, testing, and governance activities.

1.1 Purpose

This Software Requirements Specification (SRS) document defines the functional and non-functional requirements for enhancements to the CareConnect platform, an existing caregiving application. The purpose of these enhancements is to strengthen system reliability, security enforcement, deployment safeguards, billing correctness, and notification behavior while preserving established user workflows and system architecture.

This document serves as the authoritative reference for stakeholders, developers, testers, and reviewers involved in the design, implementation, validation, and evaluation of the proposed enhancements. It provides a clear basis for requirement traceability, change control, and future cohort continuity.

The SRS is intended to support implementation and verification activities without redefining the underlying system, and it assumes familiarity with the baseline CareConnect platform and prior project documentation.

1.2 Document Conventions

This SRS follows the IEEE 830 standard:

- The word “shall” designates a binding requirement.
- Informative or explanatory text avoids normative language.
- Requirements are uniquely identified using structured identifiers aligned to their respective sections (e.g., SAE-###, CDG-###, SBE-###, NRR-###, NFR-###).
- Section numbering reflects hierarchical organization.
- Acronyms and specialized terminology are defined on first use or referenced from the project glossary.

1.3 Intended Audience and Reading Suggestions

1.3.1 Intended Audience

This document is intended for the following audiences:

- **Product Owner / Stakeholders**

  To review and validate that the documented requirements accurately reflect the intended enhancements, priorities, and constraints of the CareConnect platform.

- **Developers / Implementers**

  To understand the functional scope, constraints, and required behaviors of the system enhancements. Developers should focus primarily on Sections 2 and 3.

- **Quality Assurance / Testers**

  To derive test cases and validation criteria from the defined requirements. Testers should focus on Section 3 (System Features) and any applicable non-functional requirements.

- **Professors / Instructors / Evaluators**

  To assess completeness, clarity, traceability, and adherence to SRS standards within an academic or capstone context.

- **Future Cohorts / Maintainers**

  To understand the rationale, scope, and boundaries of the enhancements for continuity and future evolution of the system.

1.3.2 Reading Suggestions

- Readers seeking an overview of the system and enhancement context should begin with Sections 1 and 2.
- Readers focused on implementation or validation should proceed directly to Section 3 after reviewing the scope in Section 1.4.
- Readers concerned with constraints, assumptions, and dependencies should review Sections 2.5 through 2.9.

1.4 Product Scope

This section defines the scope of the CareConnect enhancements described in this SRS. The scope focuses on strengthening system behavior and governance within an existing application, rather than introducing a new product or redefining core caregiving workflows.

1.4.1 In-Scope

The following areas are included within the scope of this enhancement effort:

- Enforcement of automated software quality and security checks
- Strengthening CI/CD behavior to prevent promotion of failing or insecure builds
- Platform-compliant subscription billing behavior
- Reliable, auditable notification and reminder handling
- Governance mechanisms supporting traceability, change control, and future cohort continuity

1.4.2 Out-of-Scope

The following areas are explicitly excluded from this scope unless otherwise approved through formal change control:

- Redesign of core patient or caregiver workflows
- Introduction of new user roles beyond those already defined
- Changes to underlying data models unrelated to the enhancements
- Expansion of third-party integrations beyond those already in use
- Major user interface redesign unrelated to enforcement or reliability concerns
- Any functionality, behavior, or modification not explicitly identified as in scope within this SRS

1.4.3 Business Justification

These enhancements are driven by the need to ensure system reliability, security, and correctness in a healthcare-sensitive environment. Strengthening enforcement and governance mechanisms reduce operational risk, supports compliance expectations, and ensure continuity and maintainability for future development cohorts.

1.4.4 Affected Subsystems

The enhancements described in this document may impact the following subsystems:

- Continuous Integration and Continuous Deployment pipelines
- Static analysis and security scanning tooling
- Notification and reminder services
- Subscription billing and payment handling components
- Supporting documentation and governance artifacts

This enhancement effort introduces governance, enforcement, and reliability improvements without redesigning the core CareConnect caregiving workflows. Changes are expected to primarily affect CI/CD behavior, automated validation, notification delivery handling, and subscription enforcement logic. Existing user-facing workflows remain functionally consistent unless explicitly modified by requirements in Section 5.

1.4.5 Nature and Deployment of Enhancements

The enhancements defined in this Software Requirements Specification are **enforcement-focused and corrective in nature**, and are intended to strengthen reliability, security, compliance, and governance within the existing CareConnect platform. These enhancements **do not introduce new caregiving workflows, user roles, or functional feature sets**, nor do they redesign existing user-facing interactions except where required for compliance or enforcement. The scope of change is primarily **behavioral and constraint-based**, affecting how the system enforces quality gates, security validation, billing compliance, and notification reliability under defined conditions. Existing user workflows remain functionally consistent during normal operation; however, **system behavior may change when enforcement conditions are met**, such as build failures, security findings, billing status changes, or notification delivery failures.
All enhancements described in this document are required to be **validated locally by developer**s and **automatically validated through Continuous Integration pipelines** prior to promotion. Enhancements shall be deployed through controlled CI/CD processes and validated in **non-production environments** before any demonstration or approval-based release. Promotion of enhancements without successful validation is prohibited. Production deployment is out of scope unless explicitly approved through formal change control.

1.5 References

The following documents, standards, and resources provide background information and context for the requirements specified in this SRS:

Amazon Web Services, Inc. (2024). *Amazon Simple Notification Service (SNS) documentation*. https://aws.amazon.com/sns/

Apple Inc. (2024). *Apple Pay documentation*. https://developer.apple.com/documentation/passkit/apple-pay

CareConnect Project Team. (2025). *CareConnect project business needs statement*.

CareConnect Project Team. (2025). *Proposed change request form (CC-CR-001).*

GitHub, Inc. (2024). *GitHub Actions documentation*. https://docs.github.com/en/actions

Google LLC. (2024). *Google Pay API for Android: Overview*. https://developers.google.com/pay/api/android/overview

Google LLC. (2024). *Google Pay API for web: Overview*. https://developers.google.com/pay/api/web/overview

Google LLC. (2024). *Google Play subscription policies*. https://support.google.com/googleplay/android-developer/answer/9900533

IEEE Computer Society. (2018). *Systems and software engineering—Life cycle processes—Requirements engineering (IEEE Std 29148-2018).* https://doi.org/10.1109/IEEESTD.2018.8559686

University of Maryland Global Campus. (2025). *Software requirements document: CareConnect* [Capstone contributors: T. Adams, M. Lord, M. Maloney, A. Osterneck, M. Pingel, L. Rocha, & S. Wagner].

2. Overall Description

CareConnect is a mobile-first caregiving application that supports coordination between patients and caregivers through reminders, notifications, task management, and related care support functions. This project represents a **modification of an existing system**, not the development of a new platform.

The objective of this enhancement effort is to improve system reliability, enforce software quality and security standards, and ensure correct behavior across deployment, billing, and notification workflows. Existing functionality, user roles, and interaction patterns are preserved unless explicitly modified by approved requirements in this document.

CareConnect operates in a healthcare-sensitive environment and must emphasize controlled access, predictable behavior, and traceability of changes. The system is delivered through mobile and web interfaces supported by cloud-based services and automated deployment processes.

2.1 Product Perspective

CareConnect is an enhancement to an existing caregiving platform and is not a standalone or replacement system. The platform already supports patient–caregiver interaction through mobile and web interfaces, and this iteration builds upon that established foundation.

The system operates within a cloud-hosted environment and integrates client-facing applications with backend services responsible for business logic, data persistence, notifications, and billing. Existing repositories, CI/CD pipelines, and operational practices remain in place and are leveraged rather than replaced.

This enhancement effort focuses on improving enforcement and correctness within the current system—specifically around software quality controls, security validation, deployment safeguards, and operational reliability—while maintaining compatibility with existing workflows and user expectations.

2.2 Product Features

At a high level, the CareConnect platform provides caregiving support through coordinated reminders, notifications, and task management between patients and caregivers. This

enhancement effort does not redefine core functionality, but instead strengthens the correctness, reliability, and governance of existing features.

Key product feature areas relevant to this SRS include:

- Care Coordination and Reminders

  Scheduling, reminders, and notifications supporting patient care and caregiver oversight.
- Notification and Escalation Handling

  Delivery of time-sensitive alerts and escalations to appropriate users based on system rules.
- Subscription Billing and Access Enforcement

  Platform-compliant billing behavior and enforcement of access based on subscription status.
- Quality and Security Enforcement

  Automated validation of code quality, security posture, and test results prior to Deployment.
- Deployment Safeguards and Operational Reliability

  Use of controlled CI/CD processes to prevent promotion of failing or non compliant builds.

Figure 1 (Appendix B) provides a high-level interaction overview illustrating how Care Recipients, Caregivers, and system services coordinate reminders, subscription validation, and escalation behavior. The diagram contextualizes the feature areas described above by visually representing user actions, system-triggered events, and enforcement logic across application and backend components.

Detailed functional requirements, stimulus/response sequences, and enforcement behaviors for enhanced or modified functionality within these feature areas are specified in **Section 5 (System Features)**.

2.3 User Classes and Characteristics

The CareConnect system supports two primary user classes: **Care Recipient** and **Caregiver**. Differences in responsibility and access are expressed through role capabilities and caregiver subtypes rather than separate user roles.

- **Care Recipient**

  Patients are the recipients of care and represent the basic user role within CareConnect. Patients interact with the system to receive reminders and notifications, view assigned tasks and manage their own profile information. Patients have limited access to system features and do not manage other users.

- **Caregiver**

  Caregivers are users responsible for supporting one or more patients and are granted broader system access than patients. Caregivers can:

  - View the Caregiver Dashboard (mobile and web)
  - Link to one or more patients
  - Assign tasks and schedules to patients
  - Monitor patient-related information
  - Receive reminder escalations
  - Manage caregiver profile details (e.g., name, contact information, specialties, profile image)

  Caregivers may be either:

  - **Professional caregivers** (e.g., Direct Support Professionals), or
  - **Family caregivers**

  Both professional and family caregivers operate under a unified Caregiver role. Differences in access level (e.g., full vs. read-only access) are handled through permissions rather than separate roles.

- **Multi-Caregiver Support**

  The system supports a one-to-many relationship between patients and caregivers. A single patient may be associated with multiple caregivers concurrently.

- **Patient–Caregiver Linking**

  Caregivers may add existing patients to their Caregiver Dashboard by searching for a

patient using an email address and sending an invitation request. Linking may also be initiated by patients. Supported linking mechanisms include invitation-based workflows and QR code–based workflows, with approval required before access becomes active.

- **Access Control and Permissions**
  Patients have limited access to system features. Caregivers are granted additional permissions to manage care-related activities and monitoring functions on behalf of their patients. Permission levels are enforced to protect patient privacy while enabling effective caregiving.
- **Family Read-Only Access**
  Caregivers may grant selected family members limited, read-only access to specific patient data. This supports family involvement and transparency while preserving patient privacy and restricting modification rights.

All user classes and access models described above are considered **in scope**. The user classes defined in this section are referenced throughout the functional requirements in Section 5. Where requirements affect role-based access, permissions, or notification escalation behavior, the applicable user class (Care Recipient, Caregiver, or Administrator) is implied unless explicitly stated. Role-based enforcement requirements are further governed by the Security Requirements in Section 6.3 (RBAC).

2.3.1 User Class Applicability to Requirements

Unless explicitly stated otherwise, **each system feature and requirement specifies the user classes to which it applies**. Applicability is defined at the feature or requirement level to ensure clarity regarding role-based behavior, permissions, and enforcement.
Where user class applicability is not explicitly restated within a requirement, the requirement applies only to the user classes referenced in the associated feature description. Requirements affecting access control, notification escalation, billing enforcement, or administrative behavior explicitly identify the applicable user class(es) to avoid ambiguity.

No new user roles are introduced by the enhancements defined in this SRS. All requirements apply exclusively to the user classes defined in Section 2.3 and are enforced through role-based access controls as governed by the Security Requirements in Section 6.3.
The enhancements described in this document do not introduce new caregiving workflows or alter existing user interaction patterns during normal operation. User-facing behavior remains consistent with the baseline CareConnect platform. However, **conditional behavior changes may occur under enforcement, failure, or compliance conditions**, such as failed validations, subscription state changes, or notification escalation events. These changes are intentional and are designed to enforce correctness, reliability, and compliance rather than expand functionality.

2.4 Operating Environment

This section describes the environment in which the enhanced CareConnect system is expected to operate. The requirements in this section define the platforms, infrastructure, and conditions necessary to support reliable system behavior. These operating environment requirements describe baseline conditions required to validate the enforcement enhancements defined in Sections 5 and 6 and do not redefine the core CareConnect platform architecture.

2.4.1 Hardware Requirements

OS-1: The CareConnect system shall support mobile devices capable of running modern iOS or Android operating systems.

OS-2: The CareConnect system shall support standard desktop or laptop devices for web-based access.

OS-3: The CareConnect system shall operate on cloud-hosted server infrastructure supporting backend services.

2.4.2 Software Requirements

OS-4: The CareConnect system shall support mobile operating systems including iOS and Android, with supported versions defined by platform policy.

OS-5: The CareConnect system shall support current versions of major web browsers.

OS-6: The CareConnect system shall rely on backend services and APIs supporting application functionality.

OS-7: The CareConnect system shall utilize Continuous Integration / Continuous Deployment (CI/CD) tooling for build and deployment automation.

2.4.3 Network Requirements

OS-8: The CareConnect system shall require reliable internet connectivity for mobile and web clients.

OS-9: The CareConnect system shall use secure communication channels employing encrypted protocols as defined in Section 6 Security Requirements.

OS-10: The CareConnect system shall require network availability sufficient to support real-time notifications and backend interactions.

2.4.4 System Environment

OS-11: The CareConnect system shall operate within a cloud-hosted backend environment supporting application services.

OS-12: The CareConnect system shall maintain separate environments for development, testing, and deployed demonstration or validation.

OS-13: The CareConnect system shall utilize centralized logging and monitoring to support operational visibility.

2.4.5 Environmental Constraints

OS-14: The CareConnect system shall operate in a healthcare-sensitive environment requiring conservative failure handling and data protection.

OS-15: The CareConnect system shall comply with platform policies affecting billing, notifications, and security behavior.

OS-16: The CareConnect system shall depend on external services for billing, notifications, and cloud infrastructure.

Deployment of enhancements shall follow a controlled promotion model using separated environments (development, testing, and demonstration). Enhancements shall be validated through automated CI execution, controlled deployment to non-production environments, and instructor-approved demonstration activities. Production deployment is out of scope unless explicitly approved through formal change control.

2.5 Design and Implementation Constraints

The design and implementation of the CareConnect enhancements are subject to the following constraints. These constraints define non-negotiable boundaries that influence technical decisions and prioritization.

CO-1: The CareConnect system operates in a healthcare-adjacent context and shall emphasize data protection, conservative failure handling, and controlled access to sensitive information.

CO-2: Automated static analysis, security scanning, and test execution shall be mandatory.

CO-3: The enhancement effort shall preserve fail-closed enforcement principles such that builds not meeting defined quality and security criteria are ineligible for promotion.

CO-4: CI/CD enforcement mechanisms defined in Section 5 shall operate as mandatory governance controls and shall not be bypassed without formal approval.

CO-5: Subscription billing behavior shall comply with platform policies for supported environments and shall not be bypassed or substituted without approval.

CO-6: Enhancements shall be integrated with the existing technology stack and deployment model.

CO-7: Introduction of new core technologies shall be restricted and subject to formal change control.

CO-8: All changes to approved requirements shall follow the formal change management process and shall be documented for auditability and future cohort continuity.

2.6 User Documentation

User-facing and contributor documentation shall be provided to support correct use, setup, validation, and maintenance of the CareConnect platform.

2.6.1 User Documentation

UD-1: The CareConnect system shall provide in-application guidance such as tooltips, prompts, and contextual help.

UD-2: The CareConnect system shall provide basic user instructions for patients and caregivers covering reminders, notifications, and access behavior.

2.6.2 Technical and Contributor Documentation

UD-3: The CareConnect system shall provide README-level documentation describing environment setup, build steps, and validation procedures.

UD-4: The CareConnect system shall provide documentation supporting reproducible local and deployed environments.

UD-5: The CareConnect system shall provide guidance sufficient to enable handoff and continuity for future development cohorts.

UD-6: Documentation shall be maintained alongside the system and updated to reflect approved enhancements and operational changes.

2.7 Assumptions

The following assumptions apply to the CareConnect enhancements described in this document. These assumptions inform scope, design decisions, and implementation expectations.

AS-1: The baseline CareConnect platform and its core caregiving workflows are stable and available for enhancement.

AS-2: Users have access to modern mobile devices or web browsers capable of running the supported application versions.

AS-3: Users have reliable network connectivity sufficient to receive notifications and interact with cloud-hosted services.

AS-4: Platform policies governing billing, security, and deployment remain stable during the development period.

AS-5: Third-party services required for billing, notifications, and cloud infrastructure remain available and supported.

AS-6: Future development cohorts will rely on this document and accompanying documentation for continuity and maintenance.

2.8 Dependencies

The CareConnect enhancements depend on the availability, stability, and correct operation of the following external systems and services. These dependencies must remain supported for the system to function as intended.

DP-1: The CareConnect system shall depend on cloud infrastructure services providing hosting, storage, logging, monitoring, and identity management for backend components.

DP-2: The CareConnect system shall depend on external notification services responsible for delivering push notifications and other system alerts to mobile and web clients.

DP-3: The CareConnect system shall depend on platform-provided billing and subscription services used to manage subscriptions and enforce access based on payment status.

DP-4: The CareConnect system shall depend on Continuous Integration and Continuous Deployment (CI/CD) tooling used to build, test, scan, and deploy application components.

DP-5: The CareConnect system shall depend on source code repositories hosting application source code, infrastructure definitions, and supporting documentation.

DP-6: Changes to any of these dependencies may require updates to system configuration, implementation, or requirements and shall follow the formal change management process.

2.9 Constraints

The following constraints define non-negotiable limits on the CareConnect enhancement effort. These constraints establish the boundaries within which all requirements must be implemented.

CN-1: All defined business needs and requirements shall be implemented. If any requirement cannot be completed, a formal justification shall be documented and carried forward.

CN-2: The CareConnect system shall operate with heightened sensitivity to reliability, security, and correctness due to its healthcare-adjacent use.

CN-3: Builds that fail tests, quality checks, or security scans shall not be deployed or promoted.

CN-4: The existing technology stack and deployment model shall be preserved unless changes are approved through formal change control.

CN-5: The CareConnect system shall comply with applicable platform policies governing billing, security, and application behavior.

CN-6: All unresolved issues, deviations, and approved exceptions shall be documented to support traceability and future cohort continuity.

3. System Features

Section 3 provides a high-level, non-normative summary of the system feature areas introduced by this enhancement effort. These features are derived directly from the approved business needs and represent enforcement, reliability, and compliance mechanisms rather than new caregiving workflows or user-facing functional features.

The feature descriptions in this section identify the scope and applicability of each enhancement area at a conceptual level. Detailed functional requirements, stimulus/response sequences, user class applicability, and enforcement behavior are defined in Section 5, which serves as the authoritative source for implementation and validation.

3.1 Static Analysis and Security Enforcement

- System-level enforcement feature. Applies to development and CI processes only.
- Automatically enforces code quality and security validation before changes are permitted to progress through the build and CI/CD pipeline.

3.2 CI/CD Gating and Deployment Safeguards

- System-level enforcement feature. Applies to build and CI/CD pipelines only.
- Blocks build promotion and deployment unless all required tests, quality checks, and security scans succeed.

3.3 Subscription and Billing Enforcement

- System and user-facing enforcement feature. Applies to Care Recipients and Caregivers.
- Controls access to CareConnect functionality based on platform-compliant subscription and payment status, enforcing billing correctness without introducing new user workflows.

3.4 Notification and Reminder Reliability

- System and user-facing reliability feature. Applies to Care Recipients and Caregivers.
- Ensures reliable delivery, acknowledgment tracking, and escalation of care-related notifications and reminders under defined conditions.

4. External Interface Requirements

This section specifies all interfaces—human, hardware, software, and communications—as well as operational, reporting, adaptation, and business-rule considerations that govern CareConnect's interaction with the outside world.

This section defines mandatory external interface requirements for the CareConnect platform. Named platforms, services, protocols, and technologies referenced in this section represent required constraints derived from approved business needs and compliance obligations, not illustrative examples.

Where external interfaces relate to build, validation, deployment, or enforcement processes, they apply to development and operational workflows only and do not introduce direct user interaction. Interfaces related to billing, notifications, or messaging may affect both system behavior and user-facing outcomes, while remaining centrally managed and enforced by the system.

CO-4.1: External interfaces shall comply with applicable security, privacy, and healthcare regulatory constraints.

CO-4.2: All external interfaces shall be auditable and traceable.

4.1 User Interfaces

A detailed discussion of the CareConnect UI/UX wireframes can be found in the CareConnect Technical Design Document.

UI-1: The Caregiver Dashboard shall display tasks that are due, alerts, logs, patient lists, and billing status.

UI-2: The Patient Home interface shall show daily tasks, reminders, SOS access, and vital/mood indicators.

UI-3: The Admin Panel shall display subscription status, allow user management, and provide audit views.

UD-4: All user interfaces shall provide contextual guidance such as tooltips, prompts, and basic instructions appropriate to the user's role.

4.2 Hardware Interfaces

The following hardware interface items describe baseline platform integration context. Only hardware interfaces directly impacted by the enhancement scope defined in Sections 5 and 6 are normative requirements; all other hardware descriptions are provided for system context and traceability.

All hardware endpoints shall establish secure connections to CareConnect services via HTTPS/TLS v1.3 using the Amazon API Gateway. Direct LAN connections are prohibited. The API Gateway shall accommodate both REST and WebSocket protocols for request routing, authorization, and load balancing between client devices and backend microservices.

HW-1: Mobile clients shall run iOS 15+ or Android 11+ on ARM64 devices and use the camera and microphone to capture audio/video for WebRTC, QR onboarding, and local caching.

HW-2: Wearable devices shall include Fitbit Sense 2 / Versa 4, Apple HealthKit (iOS only), and Google Health Connect (Android only), and shall transmit health metrics such as heart rate, steps, and SpO_2 through authorized APIs only, requiring OAuth 2.0 or OS-native permission.

HW-3: Smart cameras (e.g., Nest Cam, Ring Cam, generic RTSP/H.264) shall stream motion and fall detection events through an on-premises RTSP gateway with HIPAA-compliant filtering.

HW-4: Alexa Smart Home devices shall allow bidirectional interaction with caregivers for reminders and alerts, using OAuth 2.0 account linking.

HW-5: Optional hardware security keys shall support FIDO2/U2F over USB or NFC for multi-factor authentication of caregivers.

4.3 Software Interfaces

CareConnect integrates with multiple platforms and third-party APIs. The following interface items represent baseline platform context. Only interfaces directly impacted by this enhancement effort are normative requirements; all others are informational and included for architectural completeness and traceability.

All software interfaces shall enforce OAuth 2.0, OIDC, or equivalent secure authentication, and use HTTPS/TLS v1.3 or higher.

SW-1: Platform billing services (Apple App Store / Google Play) shall be used for subscription purchase, renewal, and cancellation events in supported environments.

SW-2: AWS Cognito shall provide authentication and multi-factor authentication, supporting Google, Apple, and email/password login methods while enforcing HIPAA access controls.

SW-3: APNs (Apple Push Notification Service) and FCM (Firebase Cloud Messaging) shall handle push notifications; device tokens shall be stored securely, with retries using exponential backoff.

SW-4: Fitbit Web API shall be used for wearable data ingestion, supporting OAuth 2.0 and scopes for activity, heart rate, and SpO_2, refreshed every 8 hours.

SW-5: Apple HealthKit shall provide wearable data ingestion for iOS devices only; explicit user consent is required per metric, with local caching if offline.

SW-6: Google Health Connect shall provide wearable data ingestion for Android devices, requiring the Health Connect app and granular data sharing permissions.

SW-7: Google Nest API shall provide smart-camera event streams using REST/HTTPS and WebRTC (ICE/STUN/TURN), requiring Google OAuth and event filtering before persistence.

SW-8: Alexa Smart Home Skill API shall enable voice assistant interaction via REST/HTTPS with OAuth 2.0; Amazon certification and HIPAA-compliant skill development are required.

SW-9: OpenFDA API shall provide a medication reference database using REST/HTTPS. Public API access and NDC lookup shall be supported, with caching for offline queries.

SW-10: Jitsi API shall enable telemedicine video and audio calls via JWT-secured WebRTC; end-to-end encryption and HIPAA-compliant deployment shall be maintained.

SW-11: LLM Gateway (DeepSeek API) shall provide AI assistant queries; all PHI shall be de-identified before requests, with a <2 second latency target.

SW-12: AWS S3 shall be used for media and report storage via HTTPS (signed URLs); pre-signed PUT URLs, SSE-KMS encryption, and 30-day retention for exports shall be enforced.

SW-13: AWS EventBridge shall handle JSON-based batch job scheduling; nightly analytics aggregation shall be logged via CloudWatch, with failure monitoring and reporting.

4.4 Communications Interfaces

All external and internal communication between CareConnect components shall be encrypted and secured. All traffic shall use HTTPS/TLS 1.3 with HSTS headers enabled to prevent protocol downgrades. Mutual TLS and signed JWT tokens shall be used for service-to-service authentication. Latency, reliability, and retry behaviors are governed by the Non-Functional Requirements defined in Section 6.

CI-1: Public API Gateway shall handle all client–server CRUD and authentication calls via REST/JSON over HTTPS port 443 using TLS 1.3.

CI-2: Real-time update channels shall use WebSocket over WSS/TLS 1.3 on port 443 for live dashboard metrics and notifications.

CI-3: WebRTC Media channels shall use DTLS-SRTP via TURN over UDP ports 1024–65535 with end-to-end encryption for audio/video calls between patients and caregivers.

CI-4: Billing event notifications shall be received via platform billing provider APIs and verified prior to updating subscription state.

CI-5: AWS SNS shall deliver push notifications to mobile devices via HTTPS on port 443 with TLS 1.3, applying exponential backoff for retries.

CI-6: LLM Gateway (DeepSeek API) shall handle AI assistant queries over REST/HTTPS on port 443 with TLS 1.3; all PHI shall be stripped before transmission.

CI-7: Monitoring and logging traffic shall use HTTPS (CloudWatch, Prometheus) on port 443 with TLS 1.3 for telemetry, metrics, and audit log exports.

CI-8: Admin Console access shall be via HTTPS on port 443 with TLS 1.3 and role-based access control (RBAC) for authorized staff.

4.5 Operations

CareConnect shall integrate with operational monitoring, logging, backup, and deployment infrastructure necessary to support secure and reliable system behavior. Operational mechanisms defined in this section describe external infrastructure interfaces and integration behavior. Performance targets, retention timelines, recovery objectives, and service-level thresholds are defined in Section 6 (Nonfunctional Requirements).

OP-1: The system shall emit structured operational logs to centralized logging services and integrate with monitoring and alerting systems for metrics collection and threshold-based notifications. Operational log retention duration shall be governed by Section 6 retention requirements.

OP-2: The system shall integrate with managed database backup and object storage versioning services to support restoration and data recovery procedures.

OP-3: The system shall support integration with disaster recovery infrastructure enabling environment restoration in secondary regions using infrastructure-as-code definitions.

OP-4: The system shall support controlled deployment mechanisms that enable staged rollout, rollback, and version management without disrupting authenticated user sessions.

OP-5: The system shall integrate with automated resource-scaling mechanisms to dynamically adjust compute capacity based on utilization metrics.

OP-6: The system shall integrate with incident management workflows to support detection, triage, notification, and documentation of operational incidents.

4.6 Reporting Requirements

CareConnect shall generate and deliver structured reports to authorized users and system roles. Reporting interfaces shall enforce secure generation, storage, transmission, and access control consistent with applicable security and privacy requirements defined in Section 6.

RR-1: The system shall provide a caregiver adherence dashboard report in PDF format, available on demand via the application interface, with a default monthly view.

RR-2: The system shall provide a raw activity log export in CSV format, available on demand with selectable date ranges.

RR-3: The system shall provide a billing status report for authorized financial roles in CSV format, delivered on a scheduled basis.

RR-4: The system shall provide an audit trail extract in JSON format for authorized compliance roles on a scheduled basis.

RR-5: The system shall provide a care task completion report in PDF format displaying completed versus missed tasks for a defined reporting interval.

RR-6: The system shall provide a system health summary report in CSV format for authorized operational roles, including uptime and performance metrics.

RR-7: The system shall provide an AI interaction log export in JSON format for authorized compliance review, with protected health information redacted prior to export.

4.7 Site Adaptation

CareConnect shall separate workloads across development, testing, staging, and production through environment-specific deployments. Flutter shall read environment configuration flags at build time, centrally managed via AWS SSM Parameter Store. No PHI or live billing information shall be exposed in non-production environments due to VPC isolation and IAM policies.

SA-1: us-east-1 sandbox VPC with smaller RDS (db.t3.micro), mock APIs, platform billing sandbox/test credentials, and seeded dummy patient/caregiver data.

SA-2: us-east-1 isolated VPC running automated regression suites; fake PHI, nightly CI/CD deployments.

SA-3: us-east-1 isolated VPC mirrors production infrastructure; test credentials for APIs; whitelisted client IPs; UAT cycles executed.

SA-4: us-east-1 with backup in us-west-2; autoscaling enabled; multi-AZ for MSK; HIPAA/GDPR audit logs; Cloud Map rules; cross-region DR replication.

SA-5: us-west-2 hot-standby VPC; active/passive failover; RTO <4 hours; Terraform redeployment; PHI encrypted with KMS.

4.8 Business Rules

All CareConnect components, including operations, subscriptions, security, and compliance, shall adhere to defined business rules, ensuring access control, data security, and HIPAA/GDPR compliance.

BR-1: A patient must maintain an ACTIVE subscription to add more than one caregiver.

BR-2: Each patient is billed immediately at sign-up; subsequent charges recur monthly at 00:00 UTC.

BR-3: If a subscription enters SUSPENDED status, new calls cannot be initiated; existing patient data remain view-only.

BR-4: PHI shall be stored only in HIPAA-eligible AWS services and encrypted with KMS-managed keys (AES-256).

BR-5: Wearable data older than 365 days shall be archived to S3 Glacier; retrieval requires admin approval.

BR-6: A notification is considered delivered when FCM/APNs return a success response; retries follow exponential backoff.

BR-7: PDF and CSV exports may only be generated by caregivers with PRIMARY role; delegates require explicit permission.

BR-8: Delegates may not change patient subscription status; only Primary caregivers or Admins may authorize subscription changes.

BR-9: AI/LLM responses are strictly advisory; risky queries must be routed for clinician review before release to patients.

BR-10: All PHI audit logs must be retained for a minimum of 7 years for HIPAA compliance.

BR-11: Integration with wearable or smart home devices requires explicit caregiver/patient consent before activation.

5. System Features

This section specifies the functional requirements for new or modified system features introduced by this enhancement effort. Only features added or modified are documented here. Unless explicitly stated otherwise, all requirements in this section are enforced using a fail-closed (block-by-default) model. When a required validation, check, or condition fails, the system shall prevent progression, access, or execution rather than allowing partial, degraded, or undefined behavior.

User class applicability for each feature or requirement is explicitly stated at the feature level. Where individual requirements do not restate applicability, they inherit the user class applicability defined by the associated feature description in this section.

The diagrams provided in Appendix B visually reinforce the stimulus/response sequences, state transitions, and enforcement logic described throughout this section.

Priority levels used in this section are defined as follows:

- **Critical** – Mandatory enforcement required; failure to implement or comply blocks deployment or operation.
- **High** – Mandatory enforcement required; failure materially degrades reliability, security, or compliance and must be addressed before release.
- **Medium** – Required behavior that may be deferred only with documented justification and approval.
- **Low** – Optional or improvement-oriented behavior that does not block release.

Qualitative terms such as “reliable,” “secure,” or “timely” used within this section are governed by the measurable performance, reliability, and security constraints defined in Section 6 unless explicitly quantified within an individual requirement.

5.1 Static Analysis and Security Enforcement

5.1.1 Description and Priority

This feature enforces automated static analysis, dependency vulnerability scanning, and security validation to ensure that source code meets defined quality and security standards before it progresses through the development lifecycle. The objective is to prevent insecure, unstable, or policy-violating code from entering protected branches or downstream CI/CD pipeline stages.

This feature applies to development and Continuous Integration (CI) processes only and does not introduce direct user-facing behavior.

Priority: Critical

5.1.2 Operational Rules

This subsection defines the execution model, evaluation logic, and enforcement behavior for static analysis and security validation.

Trigger Condition

Static analysis and security enforcement shall be triggered automatically upon any of the following events:

- Push to a feature branch
- Creation or update of a Pull Request (PR)
- Merge attempt into a protected branch

No manual triggering is required for enforcement.

Checks Performed

Upon trigger, the CI/CD pipeline shall execute the following categories of checks:

1. **Static Code Quality Analysis**
    - Detect syntax errors
    - Detect unused variables and unreachable code
    - Detect code smells and maintainability violations
    - Enforce formatting and style compliance
2. **Static Application Security Testing (SAST)**
    - Detect hardcoded credentials
    - Detect insecure API usage
    - Detect injection vulnerabilities
    - Detect insecure deserialization or unsafe reflection
3. **Software Composition Analysis (SCA)**

- Detect known CVEs in direct and transitive dependencies
- Identify outdated libraries with known security risks

4. **Automated Tests**
    - Unit tests
    - Integration tests (where applicable)

Evaluation Policy

The CI/CD pipeline shall:

- Execute **all configured checks**
- Collect all findings
- Evaluate severity levels before making a gating decision

The CI/CD pipeline shall **not stop at the first failure**.

All configured tools shall complete execution to provide comprehensive feedback in a single run.

Failure Policy (Fail-Closed Model)

The system shall enforce a fail-closed model:

- If any unit test fails → CI/CD pipeline status = FAILED
- If any SAST finding is High or Critical severity → CI/CD pipeline status = FAILED
- If any SCA finding is High or Critical severity → CI/CD pipeline status = FAILED
- If static analysis detects blocking-level violations → CI/CD pipeline status = FAILED

When the CI/CD pipeline status is FAILED:

- The commit shall not be eligible for merge into protected branches
- Subsequent CI/CD stages (e.g., packaging or deployment) shall not execute
- The build artifact shall not be marked deployable
- No automatic override shall be permitted.

Success Policy

If all required checks complete successfully and no blocking-level issues are detected:

- The CI/CD pipeline status shall be marked as PASSED

- The build artifact shall be marked as eligible for subsequent CI/CD stages
- Eligibility does not imply automatic production deployment
- Deployment decisions remain governed by Section 5.2 (CI/CD Gating)

Notification and Logging

Upon completion:

- CI/CD pipeline results shall be recorded in the CI system
- Detailed scan reports shall be retained as artifacts
- The Pull Request author and reviewers shall be notified via repository status checks
- Failure logs shall include:
    - Tool name
    - Severity level
    - File location
    - Description of issue

Scan results shall be retained for audit and traceability in accordance with logging policies defined in Section 4.5.

5.1.3 Stimulus/Response Sequences

Stimulus

A developer submits a code change to the approved source control repository.

Response

1. The Continuous Integration pipeline is triggered automatically.
2. Static code analysis, SAST, and SCA tools execute.
3. Automated test suites execute.
4. All tools complete execution.
5. Results are aggregated.
6. If blocking conditions are detected, the CI/CD pipeline transitions to a FAILED state.
7. If no blocking conditions are detected, the CI/CD pipeline transitions to a PASSED state and marks the build eligible for subsequent CI/CD pipeline stages.

Successful completion of all validation stages marks the build as eligible for promotion but does not automatically trigger production deployment without meeting defined promotion policies.

Figure 5 (Appendix B) illustrates the gated CI/CD activity workflow. The diagram shows the sequence of static analysis, SAST, SCA, and automated test execution stages triggered upon code submission. It also depicts the fail-closed enforcement mechanism, including where blocking decisions occur and how failed conditions prevent further CI/CD pipeline progression or deployment eligibility.

System States

SAE-OS-001

The system shall enter an automated CI execution state upon code submission.

SAE-OS-002

The system shall transition to a BLOCKED state when any defined blocking condition is detected.

5.1.4 Functional Requirements

SAE-001

The system shall automatically execute static code quality analysis for each code commit and pull request event.

Pass Criteria

- All configured static analysis tools execute successfully and no blocking-level violations are detected.

Fail Criteria

- A static analysis tool fails to execute, or a blocking-level violation is detected.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED; merge eligibility shall be denied; downstream stages shall not execute.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the pull request author and reviewers of failure.

Artifacts / Evidence

- Static analysis report artifact; CI execution logs; repository status check record.

SAE-002

The system shall automatically execute Static Application Security Testing (SAST) for each code commit and pull request event.

Pass Criteria

- All configured SAST tools execute successfully and no High or Critical severity findings are detected.

Fail Criteria

- A SAST tool fails to execute, or any High or Critical severity finding is detected.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED; merge eligibility shall be denied; downstream stages shall not execute.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the pull request author and reviewers of failure.

Artifacts / Evidence

- SAST report artifact; severity summary output; CI run record tied to commit hash.

SAE-003

The system shall automatically execute Software Composition Analysis (SCA) for each code commit and pull request event.

Pass Criteria

- SCA completes successfully and no High or Critical severity CVEs are detected in project dependencies.

Fail Criteria

- SCA fails to execute, or any High or Critical CVE is detected.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED; merge eligibility shall be denied; downstream stages shall not execute.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the pull request author and reviewers of failure.

Artifacts / Evidence

- Dependency scan report; vulnerability summary; CI run record.

SAE-004

The system shall execute automated unit and integration tests as part of the CI process.

Pass Criteria

- All required automated test suites execute and complete with zero failures.

Fail Criteria

- Any required test fails, times out, or does not execute.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED; merge eligibility shall be denied; downstream stages shall not execute.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the pull request author and reviewers of failure.

Artifacts / Evidence

- Test summary report (e.g., JUnit XML); CI test logs; repository status check record.

SAE-005

The system shall enforce a fail-closed gating policy preventing progression when blocking conditions are detected.

Pass Criteria

- When no blocking conditions exist, the CI/CD pipeline transitions to PASSED state and marks the build eligible for subsequent stages.

Fail Criteria

- A blocking condition exists and the CI/CD pipeline reports PASSED or allows progression to subsequent stages.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED; merge eligibility shall be denied; packaging and deployment stages shall not execute.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the pull request author and reviewers of enforcement failure.

Artifacts / Evidence

- CI/CD pipeline decision log; CI execution history; required status check configuration.

SAE-006

The system shall record static analysis results, security findings, dependency scan results, and test outcomes for audit and traceability purposes.

Pass Criteria

- Each CI run produces retrievable, timestamped reports associated with a build ID and commit hash.

Fail Criteria

- Required reports are missing, incomplete, or not traceable to a specific build or commit.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED due to non-compliance with traceability requirements.

Notifications

- CI/CD pipeline failure status shall be visible through CI status indicators.

Artifacts / Evidence

- Stored CI artifacts; audit log entries; CI/CD pipeline run history tied to commit hash.

SAE-007

Developers shall submit code changes through the approved source control system to trigger automated enforcement.

Pass Criteria

- All changes to protected branches occur through pull request workflow with required status checks executed.

Fail Criteria

- A change reaches a protected branch without CI enforcement running.

System Response on Fail

- Merge attempt shall be blocked by branch protection rules and the non-compliant event shall be recorded.

Notifications

- Merge denial shall be visible in the repository interface and CI status output.

Artifacts / Evidence

Branch protection configuration; pull request history; CI run linkage to merge event.

5.1.5 Severity Classification and Blocking Thresholds

This subsection defines the severity levels and enforcement thresholds used to determine CI/CD pipeline blocking behavior.

Severity Levels

Security and quality findings shall be classified into the following levels:

- **Critical**
  - Exploitable vulnerability with immediate security impact
  - Remote code execution, authentication bypass, exposed credentials
- **High**
  - Significant security risk requiring remediation before release
  - Injection risks, insecure configuration, privilege escalation paths
- **Medium**
  - Security weakness or maintainability issue requiring remediation but not immediately exploitable

- **Low**
    - Informational issues or minor quality concerns

Blocking Policy

The CI/CD pipeline shall enforce the following blocking thresholds:

- Any Critical finding → BLOCK
- Any High finding → BLOCK
- Medium and Low findings → Logged but do not block
- Failed test execution → BLOCK
- Static analysis tool execution failure → BLOCK

Aggregation Policy

- All configured tools shall complete execution before a blocking decision is made.
- The CI/CD pipeline shall aggregate results and evaluate severity collectively.

Exception Handling

- Blocking findings may not be suppressed or bypassed without:
- Documented justification
- Risk acceptance record
- Formal approval through change control

5.2 CI/CD Gating and Deployment Safeguards

5.2.1 Description and Priority

This feature enforces strict gating behavior within Continuous Integration and Continuous Deployment (CI/CD) pipelines to prevent promotion of artifacts that fail required validation checks. The objective is to ensure that only verified, compliant builds are eligible for packaging and deployment.

Priority: Critical

5.2.2 Operational Rules

Trigger Condition

CI/CD gating shall be evaluated when:

- A commit passes CI validation
- A merge to a protected branch occurs
- A deployment stage is initiated
- A release artifact is prepared

Evaluation Inputs

The gating mechanism shall evaluate:

- Results from Section 5.1 (SAE-001 through SAE-007)
- Test execution status
- Static analysis results
- Security scan results
- Dependency scan results

Gating Policy

The CI/CD pipeline shall operate under a strict fail-closed gating model:

- If any blocking condition exists → progression stops

- If no blocking condition exists → artifact becomes eligible for next stage
- The CI/CD pipeline shall evaluate all required checks before making a gating decision.

Stop Behavior

If a failure condition is detected:

- CI/CD pipeline execution halts immediately at gating stage
- Artifact is marked NOT DEPLOYABLE
- No packaging or deployment job may execute

Manual Override Policy

- Manual override of gating enforcement is prohibited.
- Protected branch settings and CI configuration shall enforce:
- Required status checks
- Non-bypassable validation
- Restricted merge permissions

Success Policy

If all required checks pass:

- Artifact is marked ELIGIBLE FOR DEPLOYMENT
- Deployment to non-production environments may proceed automatically
- Production deployment requires explicit approval (if applicable under scope)
- Eligibility does not equal automatic production release.

Logging and Audit

The system shall:

- Record gating decision
- Record reason for failure (if applicable)
- Associate decision with build ID and commit hash
- Retain logs for audit purposes

Notification Policy

CI/CD pipeline execution status (PASSED or FAILED) shall be communicated through required repository status checks and CI system indicators associated with the triggering commit or pull request. Failure states shall be immediately visible to the pull request author and reviewers through repository interfaces. No deployment stage shall execute unless required status checks report a successful state. Additional messaging integrations (e.g., email or team communication tools) may be configured but are not required for compliance with this specification.

5.2.3 Stimulus/Response Sequences

Stimulus

A CI/CD pipeline stage responsible for packaging or deployment is reached following CI validation.

Response

1. Gating logic evaluates prior CI results.
2. If any blocking condition exists:
    - CI/CD pipeline transitions to FAILED state.
    - Packaging and deployment stages do not execute.
3. If no blocking condition exists:
    - CI/CD pipeline transitions to PASSED state.
    - Artifact marked eligible for deployment.

System States

CDG-OS-001

The system shall operate in a gated execution state prior to packaging or deployment stages.

CDG-OS-002

The system shall transition to a BLOCKED state when any defined failure condition is detected.

5.2.4 Functional Requirements

CDG-001

The system shall prevent deployment of any build that fails required automated tests defined in Section 5.1.

Pass Criteria

- All required automated test suites complete successfully prior to any packaging or deployment stage.

Fail Criteria

- A required automated test fails or does not execute and a packaging or deployment stage is triggered.

System Response on Fail

- The CI/CD pipeline shall terminate at the gating stage; the deployment job shall not execute; CI/CD pipeline status shall be marked FAILED.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the PR author and reviewers of failure.

Artifacts / Evidence

- Test summary report; CI logs; deployment job skipped/cancelled record; CI/CD pipeline run history.

CDG-002

The system shall prevent deployment of any build that fails against static analysis, SAST, or SCA enforcement thresholds defined in Section 5.1.5.

Pass Criteria

- No blocking-level findings (High or Critical severity) are present prior to the deployment stage.

Fail Criteria

- A blocking-level finding exists and the CI/CD pipeline proceeds to packaging or deployment.

System Response on Fail

- CI/CD pipeline execution shall halt at the gating stage; artifact shall be marked NOT DEPLOYABLE; deployment job shall not execute.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the PR author and reviewers of failure.

Artifacts / Evidence

- Gating decision log; scan reports; CI run record tied to commit hash; skipped deployment job record.

CDG-003

The system shall immediately halt CI/CD pipeline progression at the gating stage when a failure condition is detected.

Pass Criteria

- When a blocking condition is identified, subsequent jobs (packaging, containerization, deployment) do not execute.

Fail Criteria

- A blocking condition is detected and any subsequent CI/CD pipeline job executes.

System Response on Fail

- CI/CD pipeline status shall be marked FAILED; subsequent stages shall be automatically cancelled or skipped.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the PR author and reviewers of failure.

Artifacts / Evidence

- CI execution timeline; job dependency configuration; cancelled job records; CI/CD pipeline status output.

CDG-004

The system shall prevent manual bypass of enforced gating behavior through branch protection and required status checks.

Pass Criteria

- Protected branches enforce required successful status checks prior to merge; deployment jobs require successful CI completion.

Fail Criteria

- A merge into a protected branch occurs without required status checks passing, or a deployment executes without validated CI/CD pipeline status.

System Response on Fail

- Merge attempt shall be denied by branch protection rules, and/or deployment job shall not execute due to unmet gating conditions.

Notifications

- CI/CD pipeline/merge status indicators shall notify the PR author and reviewers when gating prevents merge or deployment.

Artifacts / Evidence

- Branch protection configuration; required status check configuration; merge attempt logs; CI workflow configuration file.

CDG-005

The system shall record CI/CD pipeline execution outcomes and failure reasons for audit and traceability.

Pass Criteria

- Each CI/CD pipeline run produces retrievable records including build ID, commit hash, execution status, and failure reason (if applicable).

Fail Criteria

- CI/CD pipeline execution results are missing, incomplete, or not traceable to a specific build or commit.

System Response on Fail

- CI/CD pipeline shall be marked FAILED due to non-compliance with traceability requirements; the logging failure shall be recorded in CI output.

Notifications

- CI/CD pipeline status and required repository status checks shall notify the PR author and reviewers of failure.

Artifacts / Evidence

- Stored CI execution logs; artifact retention records; CI/CD pipeline run history.

CDG-006

Development readiness decisions shall be based exclusively on CI/CD pipeline status indicators and required status checks.

Pass Criteria

- Packaging/deployment stages are only permitted when required checks report PASSED and no blocking conditions exist.

Fail Criteria

- A deployment is executed when required checks are FAILED, missing, or not associated with the triggering commit/build.

System Response on Fail

- Deployment stage shall be blocked or cancelled and the non-compliant attempt shall be recorded for audit review.

Notifications

- CI/CD pipeline/required-check status shall notify the PR author and reviewers that deployment readiness is not satisfied.

Artifacts / Evidence

- Required status check configuration; CI run history; deployment job conditional rules audit log of blocked deployment attempt.

5.3 Subscription and Billing Enforcement

5.3.1 Description and Priority

This feature enforces subscription validation and billing compliance to ensure that access to subscription-gated functionality aligns with verified payment status and supported platform policies. The objective is to prevent unauthorized access, billing inconsistencies, and platform compliance violations.

Subscription enforcement applies uniformly across supported platforms and shall not introduce bypass paths or inconsistent state transitions.

Priority: Critical

5.3.2 Operational Rules

Subscription enforcement shall operate using a defined subscription state model.

At minimum, subscription states shall include:

- ACTIVE
- INACTIVE
- SUSPENDED
- EXPIRED
- PAYMENT_FAILED

Access decisions shall be derived exclusively from the current subscription state.

Subscription state changes shall only occur as a result of:

- Verified payment approval
- Payment failure notification
- Explicit cancellation
- Subscription expiration
- Platform billing webhook events

Access to subscription-gated functionality shall require ACTIVE state.

Subscription validation shall occur prior to execution of any gated functionality.

All subscription state changes and billing outcomes shall be recorded for audit purposes.

5.3.3 Stimulus/Response Sequences

Stimulus

- A user attempts to access subscription-gated functionality.
- A user initiates a subscription purchase or renewal.
- A billing event occurs (payment approval, failure, cancellation, expiration).
- A platform webhook updates subscription status.

Response

1. The system evaluates the current subscription state.
2. If state = ACTIVE → access granted.
3. If payment is initiated:
    - System forwards request to supported billing platform.
    - Upon verified approval → state transitions to ACTIVE.
4. If payment fails → state transitions to PAYMENT_FAILED or SUSPENDED.
5. If subscription expires or is cancelled → state transitions to EXPIRED or INACTIVE.
6. Access restrictions are enforced immediately based on state.
7. State transitions and billing outcomes are logged.

Figure 2 and Figure 3 (Appendix B) illustrate the subscription purchase and enforcement lifecycle. Figure 2 depicts the event-driven billing validation and activation flow, including successful and failed payment handling. Figure 3 presents the subscription state transition model, showing how access eligibility changes in response to activation, renewal, suspension, expiration, and cancellation events.

System State

SBE-OS-001

The system shall validate subscription state prior to granting access to subscription-gated functionality.

5.3.4 Functional Requirements

SBE-001

The system shall evaluate the user's subscription status prior to granting access to subscription-gated functionality.

Pass Criteria

- Access to gated functionality is permitted only when subscription state = ACTIVE.

Fail Criteria

- Access to subscription-gated functionality is granted when subscription state is not ACTIVE.

System Response on Fail

- Access shall be denied; user shall be redirected to subscription or billing interface.

Notifications

- Access denial shall be communicated through application interface messaging indicating inactive subscription status.

Artifacts / Evidence

- Access control logs; subscription state record; audit log entry for denied access.

SBE-002

The system shall activate or renew a subscription only upon verified payment approval from the supported billing platform.

Pass Criteria

- Subscription state transitions to ACTIVE only after receipt of verified payment confirmation.

Fail Criteria

- Subscription state transitions to ACTIVE without verified payment confirmation.

System Response on Fail

- Subscription activation shall be blocked; state shall remain unchanged; incident shall be logged.

Notifications

- User shall receive confirmation message only upon successful activation.

Artifacts / Evidence

- Billing platform transaction record; subscription state transition log; audit trail entry.

SBE-003

- The system shall restrict access to subscription-gated functionality when a subscription is INACTIVE, EXPIRED, SUSPENDED, or PAYMENT_FAILED.

Pass Criteria

- Users in non-ACTIVE states are prevented from accessing gated functionality.

Fail Criteria

- A user in a non-ACTIVE state accesses subscription-gated functionality.

System Response on Fail

- Access shall be denied; gated functionality shall not execute.

Notifications

- User shall receive in-application notice indicating subscription restriction.

Artifacts / Evidence

- Access control enforcement log; subscription state record.

SBE-004

The system shall update subscription state promptly upon payment approval, failure, cancellation, or expiration events.

Pass Criteria

- Subscription state reflects billing event outcome prior to next access validation attempt.

Fail Criteria

- Subscription state remains outdated after receipt of billing event.

System Response on Fail

- State reconciliation process shall trigger; inconsistent state shall be logged for review.

Notifications

- User-facing confirmation or restriction messaging shall reflect updated state.

Artifacts / Evidence

- Webhook receipt log; state transition record; timestamped billing event record.

SBE-005

The system shall comply with supported platform billing policies for all subscription transactions.

Pass Criteria

- All subscription transactions are processed exclusively through approved platform billing mechanisms.

Fail Criteria

- Subscription activation occurs outside approved billing platform or violates platform policy.

System Response on Fail

- Non-compliant transaction shall be rejected and logged.

Notifications

- User shall be informed if transaction cannot proceed due to policy restriction.

Artifacts / Evidence

- Transaction routing logs; billing provider records; compliance review documentation.

SBE-006

The system shall record payment outcomes, subscription state changes, and enforcement actions for audit and traceability purposes.

Pass Criteria

- Each billing event and subscription state transition produces a retrievable audit record tied to user ID and timestamp.

Fail Criteria

- Billing event or state transition occurs without associated audit record.

System Response on Fail

- Event shall be logged as non-compliant; access decisions shall default to restricted state until reconciled.

Notifications

- Audit logging failure shall be visible in system logs for administrative review.

Artifacts / Evidence

- Audit log entries; billing event records; subscription state history.

5.4 Notification and Reminder Reliability

5.4.1 Description and Priority

This feature ensures reliable delivery of reminders and notifications and enforces defined escalation behavior when acknowledgments are not received within configured time thresholds. The objective is to prevent missed or ignored care-related notifications from resulting in unnoticed care gaps while preserving predictable and auditable system behavior.
This feature applies to both routine and time-sensitive reminders across supported platforms.

Priority: High

5.4.2 Operational Rules

Notifications and reminders shall operate using the following lifecycle states:

- GENERATED
- DELIVERED
- ACKNOWLEDGED
- UNACKNOWLEDGED
- ESCALATED
- FAILED

The system shall:

- Attempt delivery upon trigger
- Record delivery status
- Monitor acknowledgment status
- Initiate escalation if acknowledgment is not received within configured time threshold
- Record all state transitions

Retry attempts shall follow exponential backoff policy prior to escalation.
Escalation recipients shall be defined based on caregiver association and configured notification rules.

5.4.3 Stimulus/Response Sequences

Stimulus

A notification or reminder is generated due to:

- Scheduled trigger time
- Care-related event
- System condition
- User action
- Message requiring acknowledgment

Response

1. The system attempts delivery to intended recipient(s).
2. Delivery status is recorded.
3. If acknowledged → state transitions to ACKNOWLEDGED.
4. If not acknowledged within defined interval → state transitions to ESCALATED.
5. Escalation notification is delivered to designated caregiver(s) based on patient–caregiver assignment relationships and configured notification policies.
6. All transitions are logged.

Figure 4 (Appendix B) illustrates the reminder delivery and escalation workflow. The diagram demonstrates scheduled trigger events, acknowledgment capture, retry behavior with exponential backoff, and escalation to caregivers when acknowledgment thresholds are exceeded. This visual representation complements the defined reliability and escalation requirements.

System State

NRR-OS-001

The system shall operate in an active monitoring state until acknowledgment or escalation resolution occurs.

5.4.4 Functional Requirements

NRR-1

The system shall deliver reminders and notifications reliably to intended recipients.

Pass Criteria

- At least 95% of scheduled reminders are delivered within 5 seconds of scheduled trigger time under nominal operating conditions.

Fail Criteria

- Delivery rate falls below 95% within defined monitoring window under nominal conditions.

System Response on Fail

- Delivery failure shall be logged; retry logic shall initiate according to defined backoff policy.

Notifications

- Delivery success or failure shall be reflected in system logs and available to authorized caregivers.

Artifacts / Evidence

- Notification delivery logs; timestamped trigger and delivery records; performance monitoring reports.

NRR-2

The system shall record notification delivery and acknowledgment status.

Pass Criteria

- Each generated notification produces a retrievable record including timestamp, recipient, delivery status, and acknowledgment status.

Fail Criteria

- A notification is generated without corresponding delivery and acknowledgment record.

System Response on Fail

- Event shall be logged as non-compliant; system shall default to escalation logic if acknowledgment status cannot be verified.

Notifications

- Administrative users shall be able to view notification status history via authorized interface.

Artifacts / Evidence

- Notification state transition log; acknowledgment records; audit log entries.

NRR-3

The system shall initiate defined escalation behavior when a notification is not acknowledged within the configured interval.

Pass Criteria

- When acknowledgment is not received within defined time threshold, escalation notification is triggered to designated caregiver.

Fail Criteria

- Acknowledgment is not received within threshold and escalation is not triggered.

System Response on Fail

- Escalation logic shall be invoked; failure to escalate shall be logged for review.

Notifications

- Escalation notification shall be delivered to designated caregiver or recipient.

Artifacts / Evidence

- Escalation event log; acknowledgment timestamp comparison record.

NRR-4

The system shall notify caregivers or designated recipients when escalation conditions are met.

Pass Criteria

- Escalation recipient receives notification upon escalation trigger.

Fail Criteria

- Escalation condition occurs and no escalation notification is delivered.

System Response on Fail

- Retry logic shall execute; failure shall be logged.

Notifications

- Escalation notification delivered through supported notification channel.

Artifacts / Evidence

- Escalation delivery log; retry attempt records.

NRR-5

The system shall retain notification and escalation history for oversight, analytics, and compliance purposes.

Pass Criteria

- Notification and escalation records are retained and retrievable for defined retention period.

Fail Criteria

- Notification or escalation history is missing, incomplete, or not retrievable.

System Response on Fail

- Event logged as audit failure; monitoring alert generated.

Notifications

- Administrative users shall be able to access retained notification history.

Artifacts / Evidence

- Stored notification logs; audit trail entries; retention configuration records

NRR-6

The system shall retry failed notification delivery attempts using exponential backoff prior to escalation.

Pass Criteria

- At least three retry attempts occur before notification is marked FAILED or escalated.

Fail Criteria

- Notification is marked FAILED or escalated without required retry attempts.

System Response on Fail

- Retry mechanism shall initiate; improper failure classification shall be logged.

Notifications

- Retry attempts shall be visible in system logs.

Artifacts / Evidence

- Retry attempt log; backoff interval record; delivery attempt timestamps.

NRR-007

The system shall provide supported application interfaces that allow users and caregivers to acknowledge or respond to notifications.

Pass Criteria

- Application interface includes acknowledgment or response controls accessible to intended users.

Fail Criteria

- Notification is delivered without available mechanism for acknowledgment or response.

System Response on Fail

- Notification shall be marked non-acknowledgeable; event logged for corrective action.

Notifications

- Interface shall visibly confirm acknowledgment submission to user.

Artifacts / Evidence

- UI design reference; acknowledgment event logs; interaction records.

6. Nonfunctional Requirements

This section defines the measurable quality attributes and operational characteristics of the CareConnect system. These requirements establish performance targets, reliability expectations, security constraints, maintainability standards, and other system qualities necessary to support healthcare-adjacent use cases and regulatory considerations.

Unless explicitly stated otherwise, these requirements apply to both mobile and web clients and the associated backend services.

Nonfunctional requirements define measurable operational thresholds and quality objectives used for monitoring and validation. Sustained or material deviation from defined thresholds shall trigger documented root cause analysis, corrective action planning, and escalation in accordance with the project governance and change management processes.

6.1 Performance Requirements

6.1.1 Concurrent User Capacity

NFR-001

The system shall support a minimum of 5,000 concurrent active users under normal operating conditions without material service degradation.

6.1.2 API Response Time

NFR-002

At least 95% of public REST API requests shall complete within 300 milliseconds under nominal load conditions.

6.1.3 Notification Delivery Latency

NFR-003

At least 95% of scheduled reminders and push notifications shall be delivered within 5 seconds of their scheduled trigger time under nominal operating conditions.

6.1.4 Payment Confirmation Latency

NFR-004

Native payment confirmation responses shall be acknowledged within 3 seconds of receipt from platform APIs.

6.1.5 Real-Time Update Propagation

NFR-005

WebSocket-based real-time updates shall propagate to connected clients within 5 seconds of event publication.

6.1.6 CI/CD Execution Duration

NFR-006

The CI/CD pipeline shall complete build, test, static analysis, and security scanning stages within 15 minutes for standard commits under normal conditions.

6.1.7 Database Query Performance

NFR-007

At least 95% of database read operations shall complete within 200 milliseconds under expected load conditions.

6.1.8 Real-Time Media Jitter

NFR-008

Real-time media streams shall maintain jitter below 200 milliseconds under nominal operating conditions.

6.2 Reliability and Safety

6.2.1 Service Availability

NFR-009

The system shall maintain at least 99.5% uptime excluding scheduled maintenance windows.

6.2.2 Graceful Degradation

NFR-010

The system shall fail gracefully when dependent external services are unavailable, ensuring no data corruption and clear user feedback.

6.2.3 Transactional Integrity

NFR-011

User data persistence operations shall be executed within transactional boundaries to prevent partial writes.

6.2.4 Service Isolation

NFR-012

Failure of one microservice shall not cause cascading failures across unrelated services.

6.2.5 Disaster Recovery Objective

NFR-013

The system shall support disaster recovery procedures with a recovery time objective (RTO) of less than four hours.

6.2.6 Incident Response Timeline

NFR-014

Sev-1 incidents shall be triaged within four hours; Sev-2 incidents within twenty-four hours; incident reports delivered within forty-eight hours.

6.3 Security

6.3.1 Data in Transit

NFR-015

All data in transit shall be encrypted using TLS 1.2 or higher.

6.3.2 Data at Rest

NFR-016

Sensitive data at rest shall be encrypted using managed encryption services.

6.3.3 Authentication

NFR-017

Authentication shall be enforced using a centralized identity provider with token-based authorization.

6.3.4 RBAC

NFR-018

Role-Based Access Control shall restrict feature access based on defined user roles.

6.3.5 Secrets Management

NFR-019

Secrets and credentials shall not be stored in source control and shall be managed using secure secret services.

6.3.6 Regulatory Alignment

NFR-020

The system shall align with HIPAA-adjacent privacy and security handling practices.

6.3.7 Vulnerability Remediation – High

NFR-021

High severity vulnerabilities shall be remediated within 14 days of identification.

6.3.8 Vulnerability Remediation – Critical

NFR-022

Critical vulnerabilities shall be remediated within 72 hours of identification.

6.3.9 Security Event Logging

NFR-023

Authentication, billing, and PHI access events shall be logged with user identifier and source IP.

6.3.10 Audit & Compliance Retention

NFR-024

Security audit logs and regulatory compliance records shall be retained for a minimum of seven years in immutable, access-controlled storage.

6.4 Maintainability

6.4.1 Code Standards

NFR-025

The codebase shall conform to standardized formatting and style rules enforced through automated tooling.

6.4.2 Reproducibility

NFR-026

Static analysis and validation steps shall be reproducible locally using documented instructions.

6.5 Scalability

6.5.1 Horizontal Scaling

NFR-027

Backend services shall support horizontal scaling without requiring application-level changes.

6.5.2 Automated Scaling

NFR-028

Backend services shall support automated scaling based on utilization metrics.

6.6 Usability

6.6.1 Platform UX Compliance

NFR-029

Payment workflows shall align with native platform user experience guidelines.

6.7 Portability

6.7.1 Multi-Environment Deployment

NFR-030

The system shall support deployment across development, staging, and production environments using infrastructure-as-code.

6.7.2 Deployment Strategy

NFR-031

Production deployments shall use zero-downtime or staged rollout strategies to prevent service interruption during updates.

6.8 Testability

6.8.1 Automated Test Coverage

NFR-032
All critical system components shall include automated unit and integration tests executed during the CI/CD pipeline.

6.9 Observability and Monitoring

6.9.1 Centralized Logging

NFR-033
All backend services shall emit structured logs to a centralized logging system including timestamp, service identifier, correlation ID, severity level, and relevant metadata.

6.9.2 Metrics Collection

NFR-034
The system shall collect operational metrics sufficient to measure compliance with defined performance, reliability, and availability requirements.

6.9.3 Alerting Thresholds

NFR-035
Automated alerts shall trigger when defined performance or availability thresholds are materially exceeded for a sustained interval.

6.9.4 Audit Log Retention

NFR-036
Security-relevant events shall be logged and retained in accordance with governance and regulatory requirements.

6.9.5 Monitoring Traceability

NFR-037

Monitoring data and alerts shall be traceable to specific deployment versions and build identifiers.

6.9.6 Operational Log Retention

NFR-038

Operational logs not classified as security audit logs shall be retained for a minimum of one year.

6.10 Nonfunctional Requirement Governance

6.10.1 Sustained Deviation Handling

NFR-039

Sustained violation of defined nonfunctional thresholds shall require documented analysis and corrective action within an approved remediation window.

6.10.2 Exception Documentation

NFR-040

Any approved exception to defined nonfunctional thresholds shall be formally documented, justified, and retained for audit purposes.

7. Future Enhancements / Deferred Requirements

The section identifies enhancements that were evaluated but intentionally deferred due to schedule, scope, resource constraints, or prioritization decisions during the current release cycle. These items are not included in the present implementation baseline and shall not be considered committed functionality unless formally approved through change control in a future release. Deferred items are documented to preserve architectural intent and support continuity for future development cohorts.

7.1 Deferred Functional Enhancements

FE-001

Advanced caregiver analytics dashboards supporting patient adherence metrics, longitudinal trend analysis, and predictive insights.

FE-002

AI-driven anomaly detection for vitals, medication adherence, and irregular care patterns.

FE-003

Multi-language notification and interface support for non-English-speaking users.

FE-004

Configurable notification escalation chains allowing caregivers and administrators to define escalation policies based on urgency and recipient role.

FE-005

Offline-first mobile workflows enabling continued access to core functionality during limited connectivity, with automated synchronization upon network restoration.

7.2 Deferred Technical Enhancements

TE-001

Replacement of Stripe-dependent billing implementation with platform-independent billing reconciliation automation.

TE-002

Multi-region AWS deployment supporting cross-region failover and disaster recovery beyond the current single-region architecture.

TE-003

Automated compliance reporting dashboards to streamline HIPAA, SOC 2, and governance reporting processes.

TE-004

Automated secrets rotation integrated with centralized secrets management services.

TE-005

Expanded observability through distributed tracing for microservices, API calls, and asynchronous background processes.

7.3 Deferred Security Enhancements

SE-001

Independent third-party penetration testing.

SE-002

Continuous Mobile Application Security Testing (MAST) for iOS and Android clients.

SE-003

Policy-as-code enforcement for infrastructure, security configuration, and compliance validation.

SE-004

SOC 2 audit readiness tooling and automation support .

7.4 Rationale for Deferral

The above enhancements were deferred to:

- Prioritize implementation of core functional, reliability, and security requirements.
- Ensure completion of critical enforcement and compliance controls before introducing expansion features.
- Avoid introducing architectural complexity that could increase risk during the current release cycle.
- Preserve a clearly documented backlog for structured future cohort continuation.

Deferred requirements may be reconsidered in future releases and shall require formal approval through the established change management process before implementation.

Appendices

Appendix A: Glossary

This glossary defines terms, acronyms, and abbreviations used throughout this Software Requirements Specification to ensure consistency and shared understanding among stakeholders, developers, testers, and future cohorts.

API (Application Programming Interface)

A defined interface that allows software components or external systems to communicate with the CareConnect platform.

Audit Log

A chronological record of system events, including authentication, access to protected data, billing actions, and administrative changes, retained for compliance and traceability.

CI/CD (Continuous Integration / Continuous Deployment)

Automated processes used to build, test, scan, and deploy software changes.

Escalation

A system-driven action triggered when a notification or reminder is not acknowledged within a defined time window.

HIPAA (Health Insurance Portability and Accountability Act)

A United States regulation governing the protection and handling of protected health information (PHI).

LLM (Large Language Model)

An artificial intelligence model used to generate responses to natural language queries, with PHI removed prior to processing.

Notification

A system-generated alert delivered to a user to communicate reminders, status changes, or care-related events.

PHI (Protected Health Information)

Individually identifiable health information that must be protected under healthcare regulations.

CI/CD Pipeline

An automated CI/CD workflow responsible for validating, building, and deploying application components.

Subscription-Gated Functionality

System features that are accessible only when a user has an active and valid subscription.

Traceability

The ability to track requirements, changes, and system behavior across development, testing, and deployment phases.

Appendix B: Analysis Models

This appendix contains analysis and design models that support understanding the CareConnect system structure, workflows, and interactions. These artifacts are intended to aid implementation, validation, and future maintenance.

The following models may be included:

- High-level system architecture diagrams
- Component interaction diagrams
- Data flow diagrams for notifications, billing, and authentication
- Sequence diagrams illustrating key workflows such as subscription enforcement and notification escalation

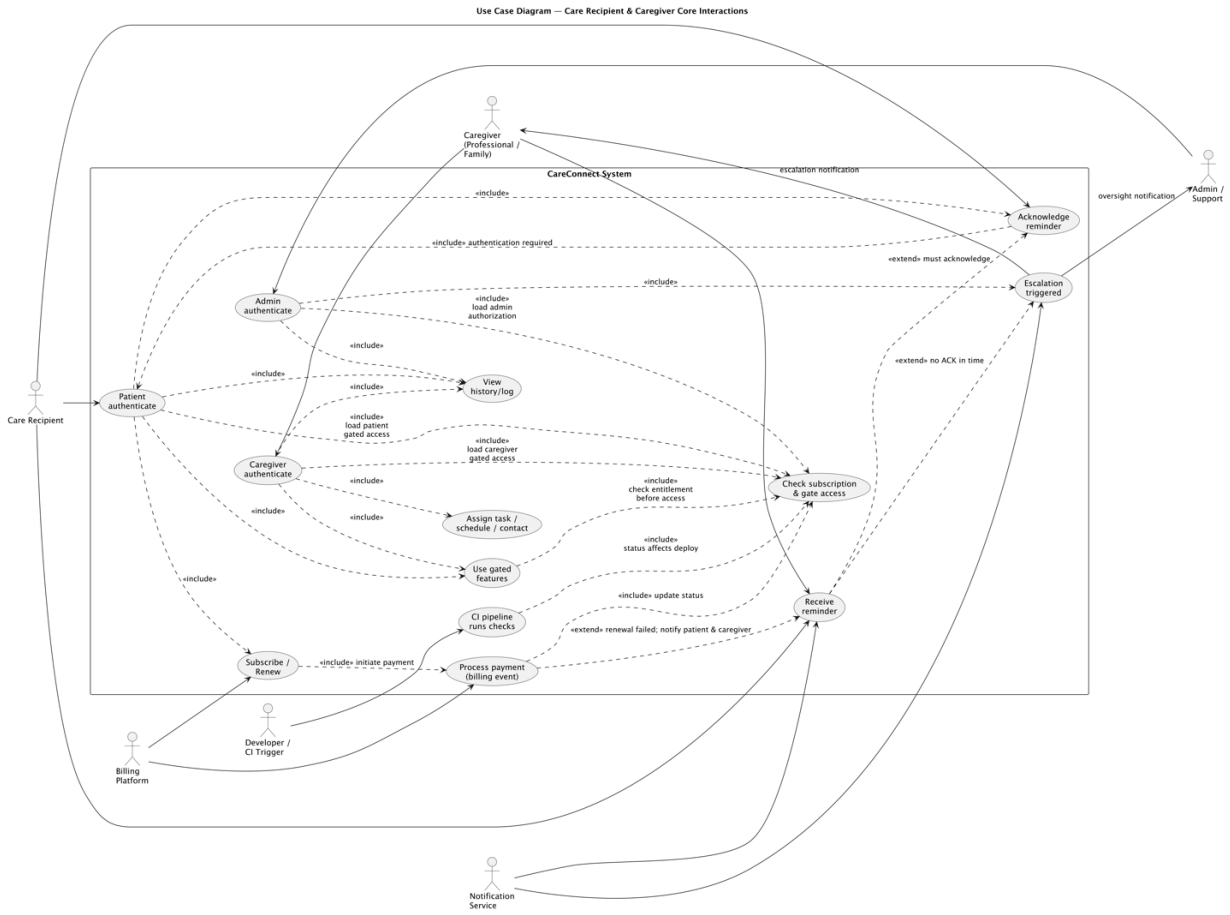


Figure 1 - Care Recipient and Caregiver Core Interaction Overview.

Note. This diagram illustrates primary interaction flows between patients, caregivers, and system services relevant to subscription validation, reminders, and escalation.

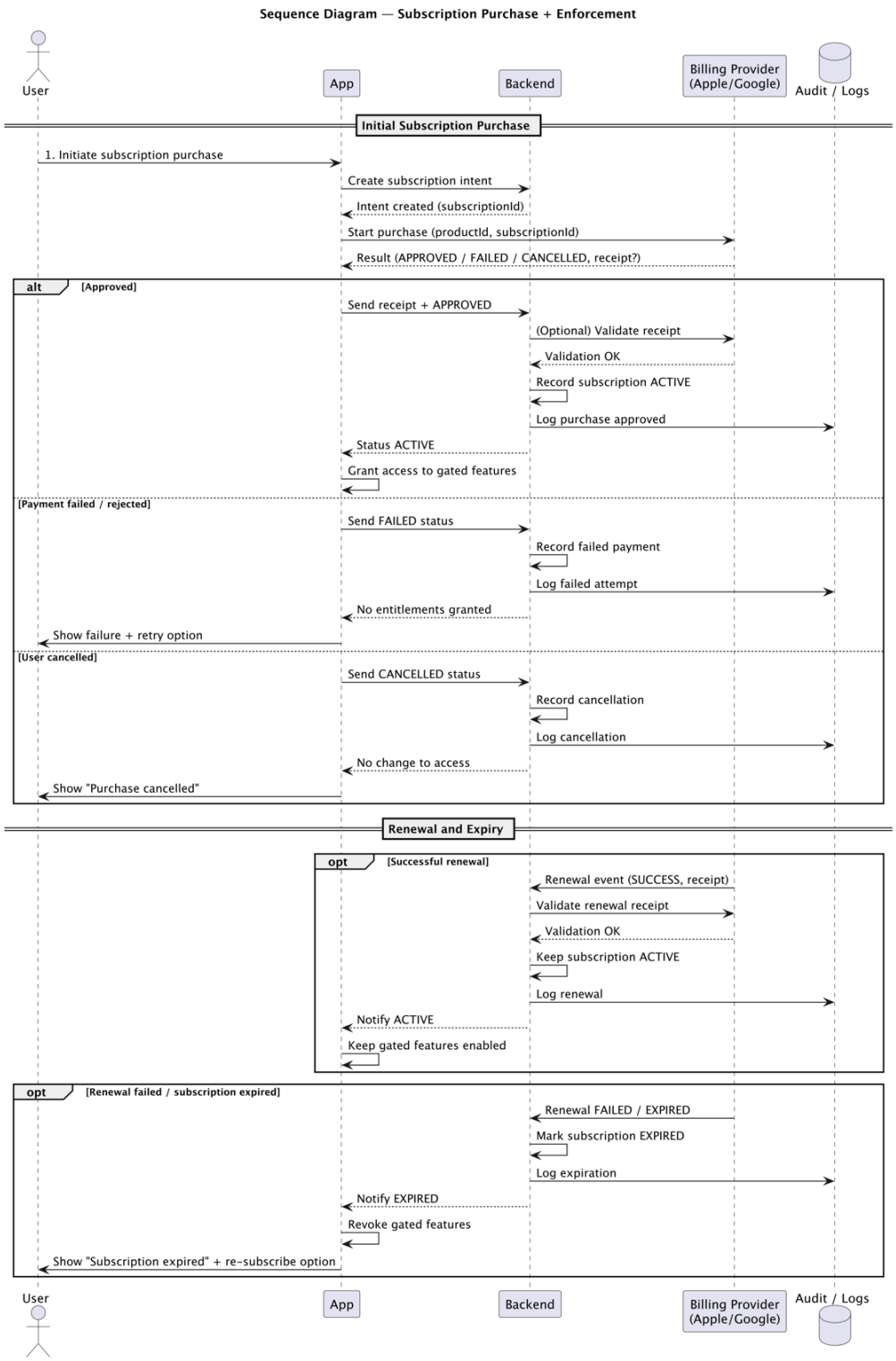


Figure 2 - Subscription Purchase and Enforcement Workflow.

Note. This diagram depicts billing validation, payment processing, subscription activation, and access enforcement logic.

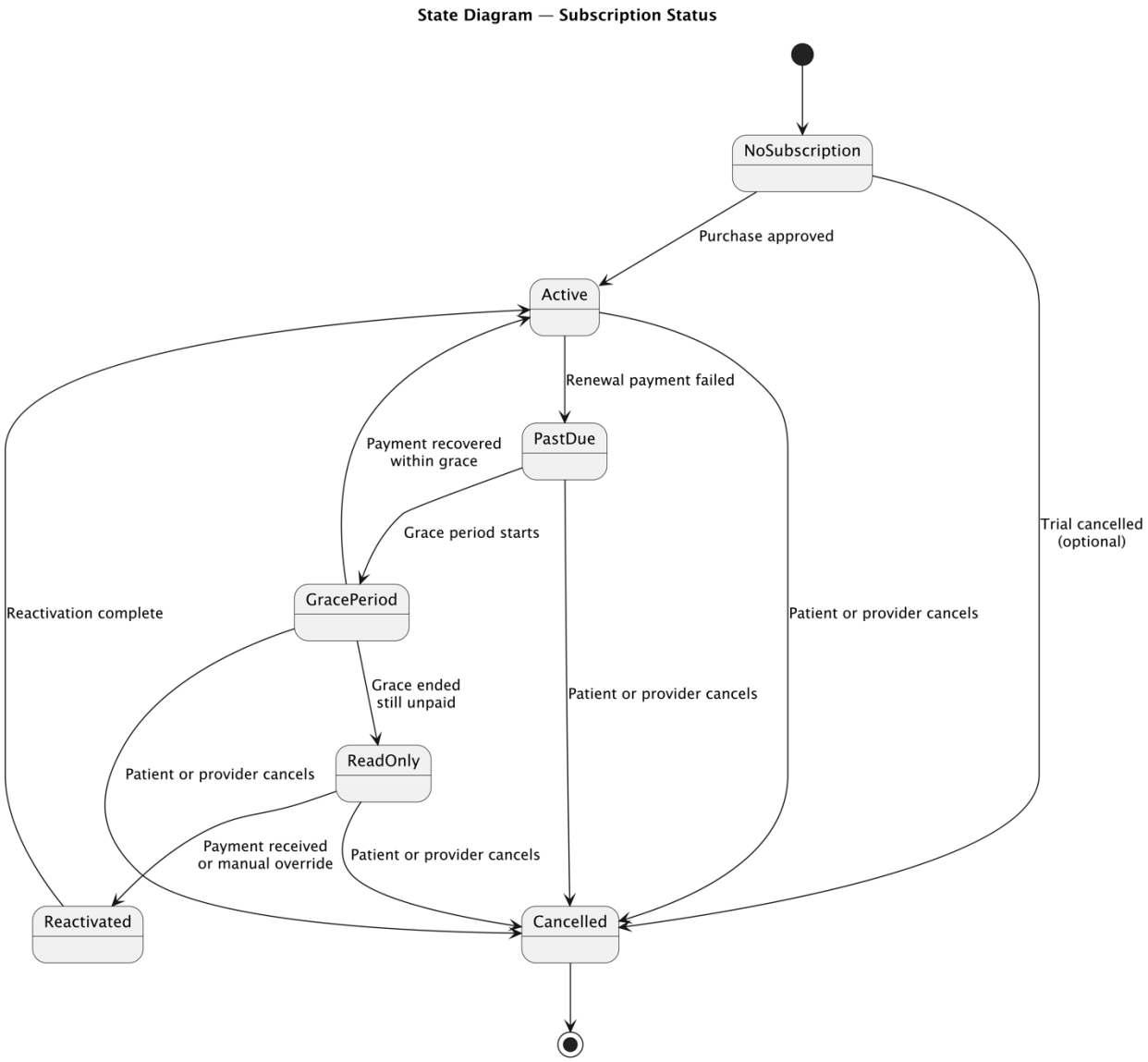


Figure 3 - Subscription State Transition Diagram.

Note. This state model shows valid subscription states and permissible transitions triggered by billing events.

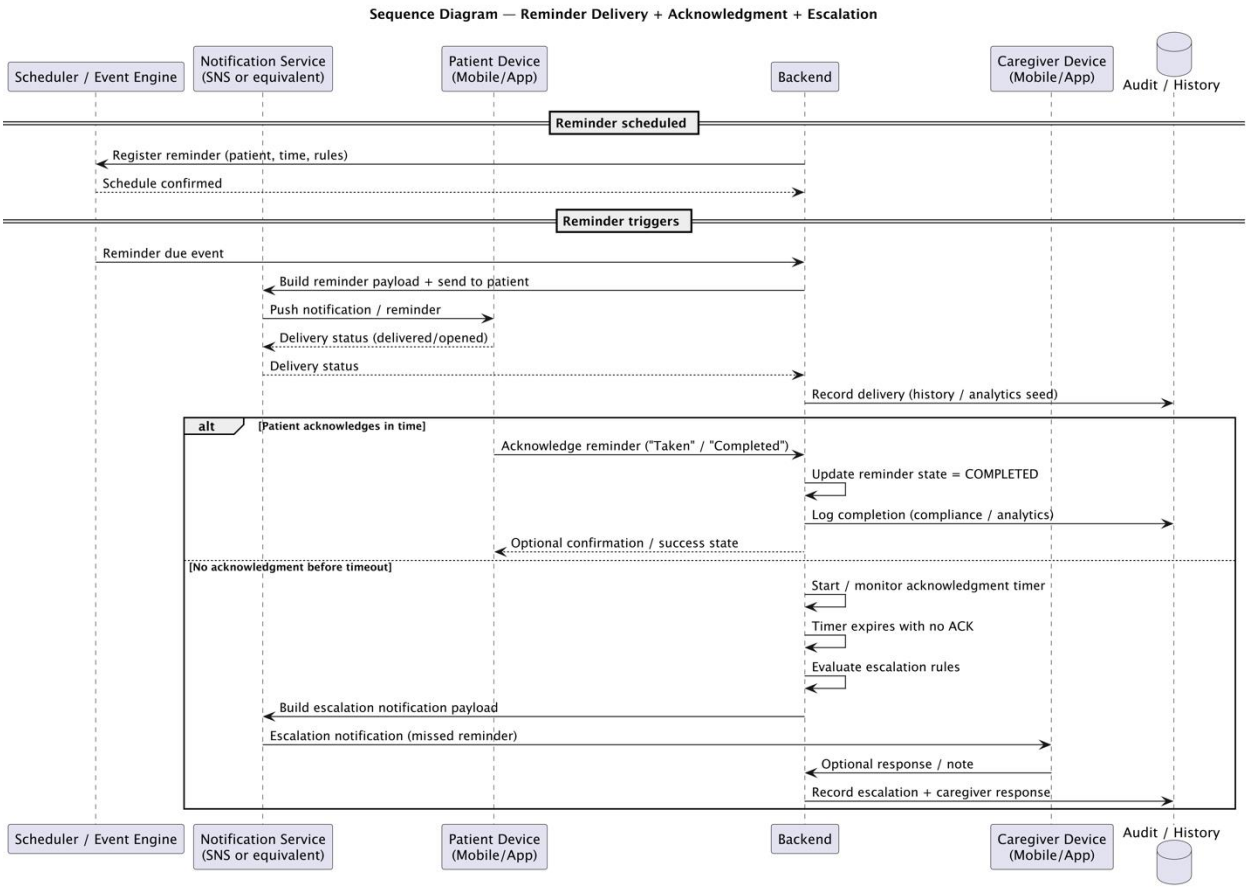


Figure 4 - Reminder Delivery, Acknowledgment, and Escalation Workflow.

Note. This sequence diagram illustrates retry behavior, acknowledgment monitoring, and escalation conditions.

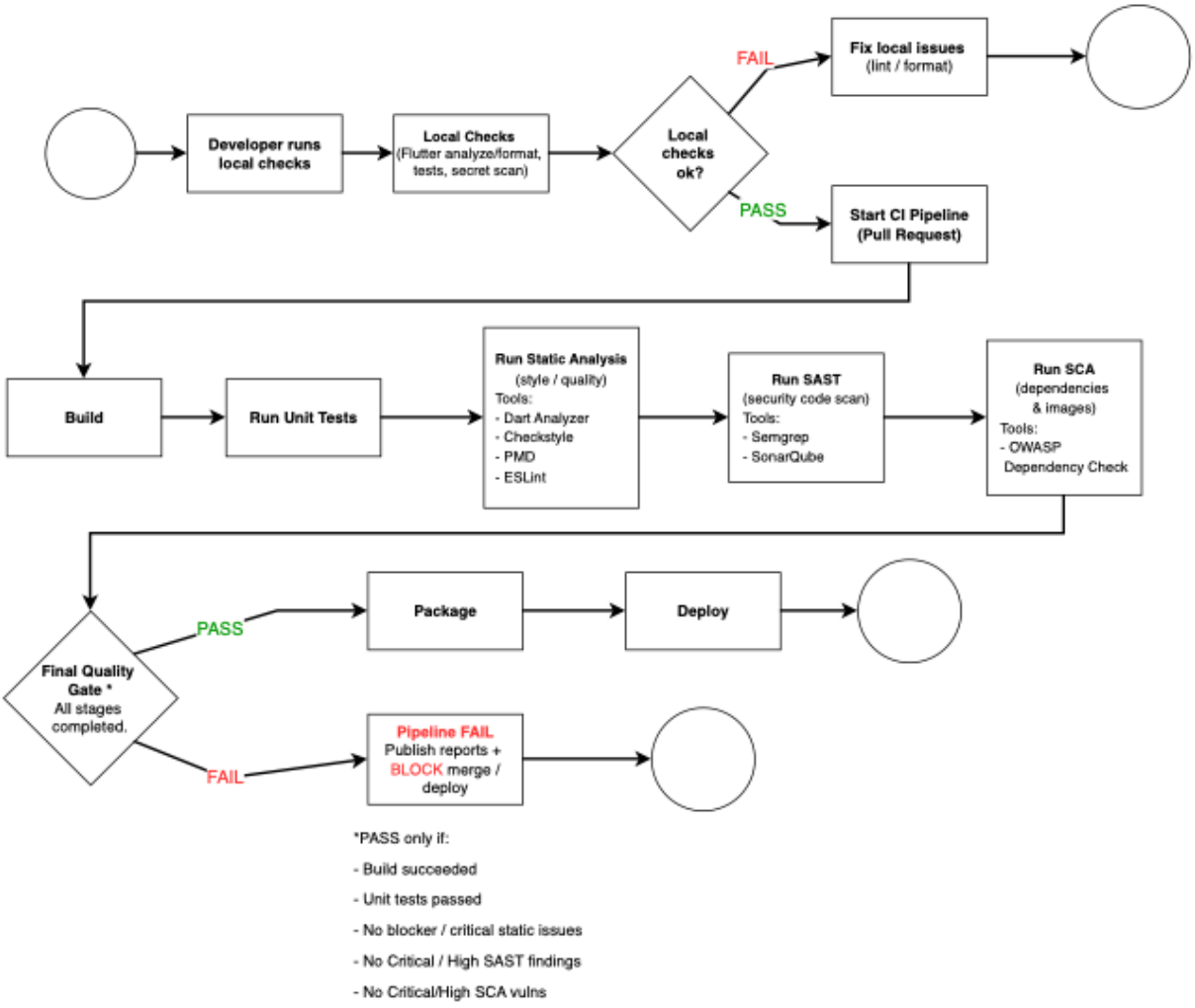


Figure 5 - CI/CD Gating and Static Analysis Enforcement Activity Diagram.

Note. This diagram shows CI/CD pipeline execution stages and fail-closed enforcement logic for code quality and security validation.

All diagrams included in this appendix shall be versioned and referenced by section number where applicable.

Appendix C

Requirements Traceability Matrix (RTM)

The Requirements Traceability Matrix (RTM) maps each Business Needs to SRS requirements and associated verification methods. This matrix ensures bidirectional traceability between defined requirements and validation artifacts in the Software Test Plan (STP). The STP version is the authoritative source.

Table C-1

Requirements Traceability Matrix (RTM).

Note. This table maps business needs to SRS requirements and associated verification methods.

BN	Business Need	Requirement ID	Type	TC	Level	Verification	Status
BN-01	Prevent unsafe builds	CDG-001	Functional	TC-001	Integration	Failing test → deployment blocked	Planned
BN-01	Prevent insecure builds	CDG-002	Functional	TC-002	Integration	Inject High/Critical finding → blocked	Planned
BN-01	Halt CI/CD pipeline on failure	CDG-003	Functional	TC-003	Integration	Introduce failing stage → jobs cancelled	Planned
BN-01	Prevent bypass	CDG-004	Functional	TC-004	Integration	Attempt protected merge → blocked	Planned

BN	Business Need	Requirement ID	Type	TC	Level	Verification	Status
BN-01	Maintain CI/CD pipeline auditability	CDG-005	Functional	TC-005	System	Inspect CI logs & artifacts	Planned
BN-01	Static analysis enforcement	SAE-001	Functional	TC-006	Integration	Verify static analysis executes	Planned
BN-01	SAST enforcement	SAE-002	Security	TC-007	Integration	Inject vuln → verify block	Planned
BN-01	SCA enforcement	SAE-003	Security	TC-008	Integration	Add vulnerable dependency → block	Planned
BN-01	Automated test execution	SAE-004	Functional	TC-009	Integration	Force test failure → CI/CD pipeline fails	Planned
BN-01	Fail-closed gating policy	SAE-005	Functional	TC-010	Integration	Introduce blocking condition → no progression	Planned
BN-01	Enforcement traceability	SAE-006	Functional	TC-011	System	Validate artifacts retained	Planned
BN-02	Enforce subscription gating	SBE-001	Functional	TC-020	System	Access without subscription → blocked	Planned
BN-02	Activate on approved payment	SBE-002	Functional	TC-021	System	Approved payment → activated	Planned

BN	Business Need	Requirement ID	Type	TC	Level	Verification	Status
BN-02	Restrict on failed payment	SBE-003	Functional	TC-022	System	Failed payment → restricted	Planned
BN-02	Update billing state	SBE-004	Functional	TC-023	System	Cancel/expire → state updated	Planned
BN-02	Record billing events	SBE-005	Functional	TC-024	System	Validate billing logs	Planned
BN-02	Prevent duplicate charges	NFR-011	Reliability	TC-025	Integration	Repeat transaction → no duplicate	Planned
BN-03	Notification latency	NFR-003	Performance	TC-030	Performance	Measure 95% ≤ 5 sec	Planned
BN-03	Transactional integrity	NFR-011	Reliability	TC-031	Integration	Mid-write failure → no partial state	Planned
BN-03	Media jitter control	NFR-008	Performance	TC-032	Performance	Measure jitter < 200ms	Planned
BN-04	Protect data in transit	NFR-015	Security	TC-040	System	Verify TLS enforced	Planned
BN-04	Protect data at rest	NFR-016	Security	TC-041	System	Verify encryption configuration	Planned

BN	Business Need	Requirement ID	Type	TC	Level	Verification	Status
BN-04	Enforce authentication	NFR-017	Security	TC-042	System	Validate token enforcement	Planned
BN-04	Enforce RBAC	NFR-018	Security	TC-043	System	Unauthorized action → blocked	Planned
BN-04	Secure secrets	NFR-019	Security	TC-044	System	Scan repo for secrets	Planned
BN-04	Regulatory alignment	NFR-020	Compliance	TC-045	System/UAT	Review privacy controls	Planned
BN-05	Maintain availability	NFR-009	Reliability	TC-050	System	Validate uptime monitoring	Planned
BN-05	Prevent cascading failures	NFR-012	Reliability	TC-051	Integration	Kill service → others stable	Planned
BN-05	Horizontal scaling	NFR-027	Scalability	TC-052	System	Scale containers	Planned
BN-05	Autoscaling	NFR-028	Scalability	TC-053	System	Trigger scale based on load	Planned
BN-05	Disaster recovery	NFR-013	Reliability	TC-060	System	Simulate region failover	Planned

BN	Business Need	Requirement ID	Type	TC	Level	Verification	Status
BN-05	Incident response timeline	NFR-014	Governance	TC-061	System	Validate response SLA records	Planned
BN-05	Centralized logging	NFR-033	Observability	TC-062	System	Verify structured logs	Planned
BN-05	Metrics collection	NFR-034	Observability	TC-063	System	Confirm metrics captured	Planned
BN-05	Alert thresholds	NFR-035	Observability	TC-064	System	Simulate threshold breach	Planned
BN-05	Audit retention (7 years)	NFR-024	Security	TC-065	System	Verify immutable retention	Planned
BN-05	Operational log retention	NFR-038	Observability	TC-066	System	Validate 1-year retention	Planned
BN-06	UX compliance	NFR-029	Usability	TC-070	UAT	Validate platform UX alignment	Planned
BN-07	Multi-environment deployment	NFR-030	Portability	TC-071	System	Deploy via IaC	Planned
BN-07	Zero-downtime deployment	NFR-031	Portability	TC-072	System	Validate staged rollout	Planned