pan-European Management of Biological toxin incidents through standaRdisAtion initiatives for Crisis response Enhancement

D3.1

Sampling devices and sample tracking - 1st Iteration

Funded by the European Union

EMBRACE is funded by the European Union's Horizon Europe Research and Innovation funding programme, Grant Agreement N° 101168322.

Lead author(s)	Sebastian Simonsen & Mees Egberts
Lead beneficiary	PRO
Status	Complete
Version	1.0
Due Date	31-03-2026
Delivery Date	27-03-2026
Dissemination Level	PU
Work Package	WP3
Task	T3.1 Sampling and forensic chain of custody
Contributors	PRO, AIR, VER
Reviewers	BIOT, AIR
Language	English
Format	DEM
Keywords	EMBRACE, CBRN, biological toxins, sample collection, chain of custody, digital chain of custody, forensic custody tracking, sample traceability, NFC, interoperability, CEN/TS 18053
Abstract	This deliverable presents the first iteration of the EMBRACE sampling and digital chain-of-custody capability under WP3 Task 3.1. It outlines the sampling approach, system architecture, operational workflow, initial validation results and current limitations. The deliverable provides the basis for the next development phase towards more robust and interoperable biological sample management in CBRN incident contexts.
Referencing this document (filename)	EMBRACE_ D3.1_PRO_Version_1.0
Disclaimer	The information and views set out in this publication are those of the author(s) and do not necessarily reflect the official opinion of the European Communities. Neither the European Union institutions and bodies, nor any person acting on their behalf, may be held responsible for the use which may be made of the information contained herein

Revision history

Version	Date	Partner	Description / Modification
0.1	01.01.2026	PRO	Table of Contents
0.2	17.02.2026	PRO	Large section contributions
0.3	25.02.2026	PRO	VER feedback addition, Section 5 reformatting
0.4	02.03.2026	PRO	Finishing touches on sections 2, 3, 4 and 5
0.5	04.03.2026	PRO	Finishing touches on sections 6, 7 and 8
0.6	09.03.2026	PRO	Executive summary and touch-ups
0.7	10.03.2026	PRO	Diagrams and screenshots were added. Ready for partner review.
1.0	27.03.2026	PRO	Added updates based on comments from internal review.

Abbreviations

API	Application Programming Interface
CAS	Chemical Abstracts Service
CBRN	Chemical, Biological, Radiological and Nuclear
CEN/TS	European Committee for Standardization / Technical Specification
CQRS	Command Query Responsibility Segregation
CTP	Custody Transfer Point
dCoC	digital Chain of Custody
DCM	Digital Custody Metadata
DMA	Differential Mobility Analyzer
DTO	Data Transfer Object
EC	European Commission
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
JWT	JSON Web Token
KPI	Key Performance Indicator
MIME	Multipurpose Internet Mail Extensions
NATO	North Atlantic Treaty Organization
NFC	Near Field Communication
OpenAPI	OpenAPI specification / interface description standard
pBDi	Universal portable detection for toxins, bacteria and viruses - AIR

RBAC	Role-Based Access Control
REST	Representational State Transfer
SOP / SOPs	Standard Operating Procedure(s)
TRL	Technology Readiness Level
UI	User Interface
UUID	Universally Unique Identifier

Table of Contents

Figures

Tables

EXECUTIVE SUMMARY

This deliverable presents the first iteration of the EMBRACE sampling and sample tracking capability developed under WP3, Task 3.1: Sampling and forensic chain of custody. It documents the methodological basis, system architecture, operational workflow, technical implementation, initial validation results, identified limitations and the planned next steps for the second development iteration.

The work addresses a central requirement in biological and CBRN incident response: the ability to collect, document, transfer and analyse samples in a manner that preserves evidential continuity and supports operational decision-making. In this context, D3.1 focuses on two complementary elements. First, it describes the sampling approach used as the reference basis for the present iteration, namely surface wipe sampling. Second, it presents the CBRN Sample Tracking System, a digital solution designed to support traceable sample registration, custody transfer and linkage to downstream analytical processes.

The selected sampling reference is based on established wipe sampling procedures suitable for the recovery of biological toxin residues from contaminated surfaces, infrastructure and equipment. The deliverable does not propose a new sampling doctrine. Instead, it situates the digital solution within recognised CBRN and forensic practice, including the requirement for documented chain of custody, clear role allocation and controlled transfer of responsibility between operational actors. In this respect, the system is intended to support, rather than replace, existing national procedures and laboratory practices.

The digital system developed in this first iteration consists of three principal components: a mobile field application, a web-based Supervisor Dashboard and a backend API. Together, these components support the registration of samples within a defined incident context, the capture of structured metadata, the binding of physical items to digital records through NFC identifiers, the grouping of samples into sealed collections, the documentation of custody transfers, and the registration of analytical activities linked to individual specimens. The architecture is designed to provide a structured and auditable operational workflow extending from field collection to custody handover and subsequent analysis.

A key feature of the solution is its alignment with CEN/TS 18053 on digital chain of custody. The system implements core concepts such as Custody Transfer Points (CTPs), structured Digital Custody Metadata (DCM), identity-linked transfer confirmation and role-based access control. From a technical perspective, the backend follows a CQRS and event-sourcing architecture, enabling immutable event logging, lifecycle reconstruction and strong traceability of all custody-relevant actions. The system also uses open technical standards, including REST APIs, OpenAPI documentation and JSON-based data exchange, in order to support interoperability and future integration with external analytical devices and other digital platforms.

The deliverable further describes the operational workflows currently supported by the system. These include incident association, sample type selection, structured metadata capture, NFC-based physical–digital binding, collection creation and sealing, analytical result registration, attachment of supporting materials, and two-party custody transfer with item-level verification. In addition, the technical implementation is documented in detail, including the software stack, data model, access control

model, event logging strategy and synchronisation mechanisms for operation in low-connectivity environments.

Initial validation activities confirm that the core concept and primary workflow components are functioning at prototype level. Functional testing verified sample registration, metadata capture, NFC-based identification, sample aggregation into collections and custody transfer recording. End-user testing provided positive confirmation of the overall concept, while also identifying several areas requiring improvement, particularly in relation to workflow clarity, user prompting, NFC interaction, interface layout and application stability.

The document is transparent regarding the current maturity level of the system. As a first operational prototype, the present iteration has several known limitations, including unauthenticated analysis endpoints, lack of persistent offline storage, limited offline support for custody transfer workflows, single-role assignment per user and the absence of certain advanced monitoring and audit features. These constraints define the scope of the current implementation and inform the priorities for the next development phase.

The second iteration will therefore focus on improving usability, resilience, security and supervisory oversight. Planned developments include clearer guided workflows, improved NFC interaction, enhanced system feedback, greater application stability and expanded Supervisor Dashboard capabilities such as incident timeline reconstruction, incident-level sample overviews, custody transfer monitoring and collection composition visibility.

Overall, D3.1 establishes the first iteration of the EMBRACE sampling and digital chain-of-custody capability and provides the technical and operational basis for its further development. It defines the main methodological, architectural and procedural elements required to support traceable biological sample management in complex CBRN contexts and lays the groundwork for the second iteration of the system.

1 INTRODUCTION

The EMBRACE project addresses the critical need for harmonised European management of biological toxin incidents through the standardisation of crisis response initiatives. This deliverable, D3.1, details the first iteration of the sampling methodologies and the digital CBRN Sample Tracking System developed within Work Package 3 (WP3).

1.1 Purpose and Scope

The primary purpose of this deliverable is to document the first development cycle and technical specifications of the EMBRACE sampling methodologies and the associated digital CBRN Sample Tracking System. As a critical output of Task 3.1, this document provides a comprehensive overview of the mechanisms used to ensure the evidential continuity of samples collected during CBRN investigations.

Surface wipe sampling has been selected as the primary reference approach for this iteration, providing an established methodology for the collection of biological toxin residues from contaminated infrastructure and equipment. While the tracking system is designed to be method-agnostic, the current scope focuses on operationalising these wipe sampling procedures through a structured digital framework. Other sample techniques like air, liquid and solid will be included in future system iterations.

The technical scope of D3.1 encompasses the following core areas:

- System Architecture: A detailed design breakdown of the multi-application environment, including the mobile field application for operational personnel, the web-based Supervisor Dashboard for incident governance, and the authoritative backend API.
- Digital-Physical Binding: The implementation of Near Field Communication (NFC) technology to establish a direct, traceable link between physical evidence items and their digital representations.
- Chain of Custody Standardisation: Alignment with European technical specifications (CEN/TS 18053) to define discrete Custody Transfer Points (CTP) and structured Digital Custody Metadata (DCM).
- Operational Resilience: The development of synchronisation mechanisms that permit the system to function in the low-connectivity or disconnected environments typical of CBRN response scenarios.

This iterative report serves as the technical baseline for the project's sampling and tracking capabilities, identifying current functional limitations to guide the refinement and security hardening scheduled for the second development phase.

1.2 EMBRACE Context and Objectives

Within EMBRACE, the sample tracking system is the digital enabling layer that connects field sampling, custody transfer and downstream analysis into a single traceable workflow. As a core output of WP3 Task 3.1, it does not replace established sampling doctrine; rather, it operationalises EMBRACE's objective of harmonising European management of biological toxin incidents by providing a

structured means to document sample collection, bind physical items to digital records, record custody transfers and preserve evidential continuity across the full operational lifecycle.

The system supports EMBRACE's 'Objective 3: Detection & identification of biotoxins' in three concrete ways. First, it strengthens operational preparedness by enabling field teams to register samples, capture structured metadata, associate samples and sealed collections with NFC identifiers, and continue operating in low-connectivity environments typical of CBRN response scenarios. Second, it strengthens evidential robustness by ensuring that each custody transition is attributable, time-stamped, verifiable and auditable, in line with the project's focus on provable chain of custody for samples that may later support confirmatory analysis, attribution or legal proceedings. Third, it supports European interoperability and harmonisation through alignment with CEN/TS 18053, including the Custody Transfer Point model and structured Digital Custody Metadata, thereby positioning EMBRACE within an emerging common European framework for digital evidence governance in CBRNE contexts.

The sample tracking system provides the backbone through which EMBRACE links sampling activities with analytical and supervisory functions. The mobile application supports field collection and custody actions, the Supervisor Dashboard provides incident-level oversight, and the backend API preserves the authoritative event history. This means that EMBRACE is not only developing sampling procedures and analytical tools in parallel, but also creating the digital governance framework needed to ensure that samples, results and custody actions remain connected, defensible and usable across the response chain.

2 SAMPLING

2.1 Wipe Sampling Procedures

Surface wipe sampling was selected as the primary sampling approach considered in this system and therefore deliverable. Wipe sampling was determined to be the most relevant for the collection of potential biological toxin residues from contaminated environments. Wipe sampling provides a practical and widely used method for detecting the presence of toxins or biological agents deposited on surfaces following an incident involving aerosol dispersion, liquid contamination or surface contact. The sample tracking system does not exclude other sample taking methods or devices but the focus for the first phase of development was using wipe sample method as reference.

The selection of surface wipe sampling as the foundational methodology for the project was driven by technical assessments from VER and complemented by inputs from consortium experts. This approach was identified as the most effective sampling protocol for the majority of biological toxin incidents, ensuring the initial development phase prioritised high-utility, validated forensic procedures.

Wipe sampling is particularly relevant in biotoxin incidents, where toxins may persist on surfaces such as equipment, infrastructure or environmental materials after the initial release event. By collecting residues from potentially contaminated surfaces, wipe samples can provide valuable material for subsequent laboratory analysis and confirmation of the presence of hazardous agents.

2.2 Provable with Chain of Custody

In incidents involving biological toxins or other hazardous biological agents, maintaining a verifiable chain of custody for collected samples is essential. Samples obtained during an investigation may be used to confirm the presence of toxic agents, support attribution assessments or serve as evidentiary material in legal or regulatory proceedings. For these reasons, it is necessary to ensure that the integrity and origin of each sample can be reliably demonstrated throughout the entire handling process.

A chain of custody refers to the documented and traceable record of the collection, handling, transfer and analysis of a sample from the point of collection through to laboratory examination and final reporting. Each stage of the process must clearly identify the responsible personnel, the time and location of the transfer and the condition of the sample at that moment. This documentation ensures that the evidential continuity of the sample is preserved and that any handling actions can be reconstructed and verified.

In the context of a biotoxin incident, maintaining a provable chain of custody is particularly important because contamination events may have significant public health, security and legal implications. Authorities must be able to demonstrate that samples were collected in accordance with recognised procedures, that they were not tampered with or contaminated during handling and that analytical results can be reliably linked to the original sampling location.

The digital custody mechanisms developed within the EMBRACE project are designed to support these requirements by providing structured documentation of sample identifiers, custody transfers and

handling events. By linking physical samples to digital records through unique identifiers and recording custody transitions within the system, the solution contributes to maintaining a transparent and auditable chain of custody throughout the operational lifecycle of a CBRN investigation.

2.3 Aerosol analysis

Aerosol sampling and analysis represent an important component of CBRN incident response, particularly in situations involving airborne biological agents or toxins. In such scenarios, rapid identification of airborne threats is essential for situational awareness, risk assessment and the protection of first responders and affected populations.

Within the EMBRACE project, aerosol detection and analysis capabilities are addressed through the development and validation of portable biosensing technologies. These technologies are designed to support rapid, on-site screening of airborne biological agents and toxins, complementing traditional laboratory-based confirmatory analysis.

The validation of these biosensing systems is described in Deliverable D3.6, *Portable biosensing devices and validation – 1st Iteration*, led by MION. In that work, the performance of the DMA and pBDi biosensing platforms is evaluated using both clean matrices and realistic environmental matrices under controlled laboratory conditions.

While the present deliverable focuses primarily on sampling procedures and the digital tracking of collected samples within the chain-of-custody framework, the aerosol analysis capabilities developed in D3.6 provide an important analytical component that complements the sampling workflow. The results obtained from portable biosensing devices can support early operational decision-making and may guide subsequent confirmatory laboratory analysis of collected aerosol samples.

Further technical details regarding the biosensing technologies, validation procedures and performance results are provided in D3.6.

2.4 CBRN Sampling and Forensic Doctrine

The sampling procedures addressed in this deliverable are aligned with established European and NATO-informed CBRN investigation doctrine (AEP-66 SIBCRA). The CBRN Sample Tracking System does not introduce new sampling methodologies; rather, it supports the documentation, traceability and governance requirements associated with recognised CBRN sampling practices.

CBRN doctrine distinguishes clearly between rapid field detection activities intended to support situational awareness and formal sampling conducted for confirmation analysis or forensic investigation. Rapid designation measures are typically undertaken during the immediate response phase to identify potential hazards and inform operational decision-making. In contrast, confirmation and forensic sampling require structured procedures, defined team roles, controlled handling and a demonstrable chain of custody to ensure evidential reliability.

Operational frameworks commonly applied in CBRN missions define specific roles, including team leadership, sampling personnel, transport functions and laboratory entities. Responsibility for samples is formally transferred between these roles at defined stages of the operational lifecycle. The CBRN Sample Tracking System reflects this structure by ensuring that each custody transition is attributable

to an identified and authorised resource, thereby reinforcing accountability across organisational boundaries.

Established sampling doctrine further requires systematic documentation of the sampling location, contextual description of the scene and, where appropriate, photographic records. Physical handling procedures typically include primary containment of the sample, secondary packaging, sealing with unique identifiers, surface decontamination prior to exit from the hot zone and controlled transfer to transport or laboratory units. These measures are designed to prevent contamination, preserve sample integrity and maintain evidential continuity.

The digital custody mechanisms implemented in the system complement these established procedures. Sample identifiers, container identifiers, seal numbers and custody acknowledgements are recorded within a structured digital environment, ensuring that the evidential history of each item can be reconstructed and audited. The system therefore reinforces recognised CBRN forensic doctrine by strengthening traceability and accountability while maintaining full compatibility with existing national Standard Operating Procedures and laboratory practices.

In this way, the deliverable situates the digital custody solution within established CBRN sampling and forensic frameworks, supporting harmonised and defensible evidence governance in line with European crisis response and investigative standards.

3 CHAIN OF CUSTODY TRACKING SYSTEM

3.1 System Architecture Overview

The CBRN Sample Tracking System is implemented as a structured, multi-application architecture composed of three principal components: (1) a mobile field application, (2) a web-based Supervisor Dashboard and (3) a central backend Application Programming Interface (API). Together, these components form an integrated digital environment supporting traceable sample registration, custody transfer and incident governance within CBRN operational contexts.

The architecture follows a clear separation of concerns. The mobile application supports operational field activities, the Supervisor Dashboard provides incident-level oversight and administrative governance, and the API functions as the authoritative backbone responsible for data integrity, authentication, authorisation and event persistence.

3.1.1 Mobile Field Application

The mobile application is designed for operational use in field environments. It enables authorised resources to document sampling activities and execute custody-related actions in accordance with defined operational roles.

Within the application, users may register individual samples and associate them with physical identifiers through NFC scanning. Each physical sample container can be linked to its corresponding digital record by scanning the NFC tag attached to the sample. This establishes a direct binding between the physical evidence item and its digital representation within the Chain of Custody system.

During sample registration, structured metadata can be assigned, including but not limited to sample type, aliquot designation, indication of control sample, solvent information and other operational attributes. This ensures that contextual and forensic parameters are documented at the point of collection in a consistent and standardised manner.

The system further supports the aggregation of multiple samples into a collection. After individual samples are registered, they may be grouped and sealed. The seal itself is equipped with an NFC identifier, which is scanned and linked to the digital collection record. This enables traceable documentation of containment and packaging procedures consistent with established CBRN forensic doctrine.

Custody transitions occurring in the field are recorded directly through the application. When a collection changes custody—for example during transfer to a transport unit—both the transferring and receiving authorised users authenticate the transfer within the application. The system records the identities of both resources, the location and a timestamp, thereby ensuring full traceability of responsibility at each Custody Transfer Point (CTP). This mechanism ensures that no transfer of custodianship occurs without explicit digital acknowledgement.

3.1.2 Supervisor Dashboard

The Supervisor Dashboard is a web-based application providing operational oversight and incident governance capabilities. It functions as the central coordination interface for supervisory personnel responsible for managing CBRN incidents.

Within the dashboard, supervisors may create and configure incidents. Each incident contains structured metadata, including geographic location, descriptive context and associated photographic documentation where applicable. Incidents serve as the primary organisational entity to which all samples, collections and custody events are linked.

Before the mobile application can be used to collect samples, operational users must associate themselves with a specific incident. This is achieved by scanning a QR code generated within the Supervisor Dashboard or by manually entering an incident code. By enforcing incident-level association prior to sampling activities, the system ensures that all subsequently recorded samples, collections and custody transfers are traceable to a defined operational context.

The dashboard provides supervisors with a comprehensive overview of incident-related activities. This includes visibility into registered samples, collections, custody transfers and role assignments. Through this interface, supervisory personnel can monitor the progression of the custody lifecycle and ensure procedural compliance.

3.1.3 Backend API

The backend API constitutes the authoritative core of the system. Both the mobile application and the Supervisor Dashboard interact exclusively with this API, which manages authentication, authorisation, event recording and data governance.

Access control is enforced through a Role-Based Access Control (RBAC) model. Each authenticated resource is assigned a role aligned with operational responsibilities, and all commands submitted to the API are validated against this role model before execution. This prevents unauthorised actions and ensures that custody operations remain consistent with defined governance rules.

The API architecture is implemented using a Command Query Responsibility Segregation (CQRS) pattern combined with event sourcing principles. All custody-related actions—such as sample registration, collection sealing and custody transfer—are recorded as immutable events within an append-only event store. This event store constitutes the authoritative evidentiary record of the system.

From this immutable event log, read models are generated to support efficient operational queries within the mobile application and Supervisor Dashboard. This separation ensures that evidentiary integrity is preserved at the source level while enabling performant access to current custody states.

Through the combination of RBAC, CQRS and event sourcing, the system provides full traceability of the lifecycle of each incident, sample and collection. Every action performed within the system is attributable to an authenticated resource and is preserved in chronological order. This architecture enables precise reconstruction of operational history and supports auditability, integrity assurance and non-repudiation in accordance with European digital Chain of Custody principles.

3.2 Technical Standards Followed

3.2.1 Digital Chain of Custody Standards (CEN/TS 18053)

The CBRN Sample Tracking System has been developed in alignment with CEN/TS 18053-1 and CEN/TS 18053-2, which define the European framework for Digital Chain of Custody (dCoC) in the

context of CBRNE evidence. These Technical Specifications establish the conceptual, structural and governance principles required to ensure traceable, verifiable and auditable custody transfer within a digital environment. Their adoption provides a harmonised European reference point for the design and implementation of the system.

CEN/TS 18053-1 defines the core concepts underpinning the custody transfer lifecycle, including the Custody Transfer Point (CTP), stakeholder roles and responsibilities, and the representation of custody events within a digital record. The CBRN Sample Tracking System applies this model by structuring each transfer of custodianship as a discrete and time-stamped CTP, requiring acknowledgement by both the transferring and receiving authorised resources. This ensures that each transition of responsibility is formally recorded and that custodianship is continuously attributable throughout the mission lifecycle.

CEN/TS 18053-2 provides guidance on Digital Custody Metadata and the associated data governance mechanisms required to maintain integrity and non-repudiation. In the present iteration, the system implements a structured metadata model capturing mission identifiers, sample identifiers, resource credentials, transfer timestamps and status indicators for each custody event. This approach ensures that the digital record is capable of supporting verification, traceability and subsequent audit.

The governance principles described in the standard have informed the system architecture. Role-based access control, authentication mechanisms and controlled validation of custody transfers are incorporated to prevent unauthorised modification of custody records. Where inconsistencies are identified during the validation of a transfer, the event is recorded and preserved within the digital log, maintaining evidential continuity and supporting subsequent review.

Although certain advanced features described within CEN/TS 18053, such as automated KPI monitoring and extended compliance analytics, are planned for further development, the current implementation is structurally compatible with these provisions. The metadata schema and workflow logic are designed to accommodate future enhancement without modification of the core custody transfer architecture.

Alignment with CEN/TS 18053 ensures that the CBRN Sample Tracking System reflects an emerging European standard for digital evidence governance in CBRNE contexts. This supports interoperability between Member States, strengthens evidentiary robustness, and positions the system within the broader European standardisation landscape for societal and citizen security.

3.2.2 Digital Custody Metadata and Data Governance Architecture

The CBRN Sample Tracking System implements a structured Digital Custody Metadata (DCM) architecture aligned with the principles defined in CEN/TS 18053-2 concerning data governance and audit within a digital Chain of Custody (dCoC). The architecture is designed to ensure that each Custody Transfer Point (CTP) is supported by verifiable, structured metadata capable of sustaining traceability, integrity and non-repudiation throughout the lifecycle of a CBRN evidence item.

At the core of the system is a formalised metadata model that captures mission identifiers, sample identifiers, collection identifiers, resource roles, timestamps and custody status indicators. Each custody transition is represented as a discrete event within the system and is persistently recorded together with the identities of the authorised custody owner and custody receiver. This structured

representation enables the reconstruction of the complete custodial history of any digital evidence item from the point of collection through transport and laboratory receipt.

The backend architecture follows a Command Query Responsibility Segregation (CQRS) pattern combined with event sourcing principles. Custody-related actions are recorded in an append-only event store, which constitutes the authoritative and immutable log of all system events. From this event stream, a separate read model database is generated to provide optimised query performance for operational use. This separation ensures that evidentiary integrity is preserved at the source level, while enabling efficient access to current custody states for users in the field and laboratory contexts.

Data governance controls are embedded within the custody transfer workflow. Validation mechanisms ensure that required metadata fields are completed before a transfer can be confirmed. Where inconsistencies or validation failures occur, the event is still recorded, and the custody state is clearly flagged, thereby preserving transparency and enabling subsequent review. This approach reflects the governance model described in CEN/TS 18053-2, where monitoring, verification and auditability form integral components of each CTP lifecycle.

The system further incorporates controlled synchronisation mechanisms to support operation in low-connectivity environments. Locally generated custody records are queued and synchronised with the central system when network connectivity is restored. All synchronised transactions are reconciled against the authoritative event store, ensuring that the digital log remains complete and chronologically consistent even when operational constraints require temporary offline operation.

Through this architecture, Digital Custody Metadata is treated as a governed digital asset rather than as a simple data record. The system provides the structural mechanisms required for integrity assurance, traceability and evidentiary audit, thereby operationalising the data governance framework foreseen in European digital chain-of-custody standards while maintaining compatibility with national forensic procedures.

3.2.3 Authentication and Role-Based Authorisation Model

The CBRN Sample Tracking System incorporates a structured authentication and role-based authorisation framework designed to ensure that custody-related actions are attributable exclusively to verified and authorised resources. This framework aligns with the principles of Authentication, Authorisation and Accounting (AAA) and supports the governance requirements set out in CEN/TS 18053 concerning resource verification within the digital Chain of Custody (dCoC).

User authentication is implemented through a secure identity management mechanism based on JSON Web Tokens (JWT). Upon successful credential validation, an authenticated session token is issued, enabling controlled access to system functions. Password credentials are protected using modern cryptographic hashing mechanisms, ensuring secure storage and resistance to compromise. All API interactions require authenticated bearer tokens, thereby preventing unauthorised system access.

Authorisation is enforced through a clearly defined role model reflecting operational responsibilities within the custody lifecycle. The system distinguishes between predefined mission roles, including documenter, transporter and laboratory resource. Each role is granted access exclusively to the functional capabilities necessary for its operational mandate. For example, field personnel may initiate

sample registration and collection activities, transport personnel may execute custody transfers, and laboratory personnel may confirm receipt and process samples. Access to protected application routes and backend endpoints is restricted according to these role assignments.

Role validation is performed both at application level and at backend service level. On the client side, protected navigation structures prevent unauthorised access to role-specific workflows. On the server side, guarded endpoints verify the role encoded within the authentication token before executing custody-related commands. This dual-layer validation ensures defence in depth and reduces the risk of privilege escalation.

Within the custody transfer process, the authentication model supports explicit attribution of custody ownership and custody reception. Each Custody Transfer Point (CTP) requires confirmation by authenticated resources, ensuring that responsibility for the evidence item is always attributable to a verified identity. Where required, discrepancies in role validation or transfer confirmation are recorded within the event log, maintaining evidentiary transparency.

By combining secure authentication, explicit role separation and backend-enforced access control, the system ensures that all digital custody events are attributable, auditable and protected against unauthorised manipulation. This model reinforces evidentiary integrity while maintaining compatibility with operational structures typically found in CBRN response and forensic investigation environments.

3.2.4 Information Integrity and Non-Repudiation Mechanisms

The CBRN Sample Tracking System has been architected to ensure that all custody-related information is protected against unauthorised alteration and that custody actions cannot be repudiated by participating resources. These objectives are achieved through a combination of architectural design choices and governance controls aligned with the integrity and traceability principles described in CEN/TS 18053.

At the core of the integrity model is an event-sourced architecture in which all custody-relevant actions are recorded as immutable events within an append-only event store. Rather than modifying or overwriting existing records, the system captures each state transition as a new event in chronological sequence. This design ensures that historical custody information remains preserved in its original form and that the complete evolution of a sample or collection can be reconstructed at any point in time. The event store therefore functions as the authoritative evidentiary log.

To support operational efficiency without compromising integrity, the system employs a Command Query Responsibility Segregation (CQRS) pattern. While the event store retains the immutable record of truth, a separate read model database is generated from the event stream to provide optimised access to current custody states. Any discrepancies between operational views and the event log can be resolved by replaying the underlying events, thereby maintaining consistency and auditability.

Non-repudiation is reinforced through authenticated custody confirmations at each Custody Transfer Point (CTP). Transfers require explicit acknowledgement by authorised resources, and each acknowledgement is recorded together with identity credentials and timestamps. Because these confirmations are preserved as immutable events, a resource cannot subsequently deny participation

in a transfer without contradicting the recorded system log. The architecture thus provides a verifiable digital history of custodianship.

Additional integrity safeguards are embedded within validation and synchronisation processes. Custody-related metadata is validated prior to acceptance, and any inconsistencies are logged rather than suppressed. In environments with limited connectivity, locally generated custody events are queued and synchronised with the central system once connectivity is restored. Synchronised transactions are reconciled against the authoritative event store, ensuring that temporary offline operation does not compromise chronological accuracy or evidential completeness.

Through these mechanisms, the system establishes a resilient digital evidence log that preserves chronological integrity, prevents silent alteration of custody data and enables full reconstruction of custody history. This approach operationalises the integrity and non-repudiation objectives central to European digital chain-of-custody standards while providing a technically robust foundation suitable for forensic and judicial scrutiny.

3.2.5 Interoperability and European Standardisation Alignment

The CBRN Sample Tracking System has been designed with explicit consideration for interoperability and alignment with emerging European standardisation efforts in the field of digital chain of custody and CBRN evidence governance. The architecture reflects the conceptual and structural principles defined in CEN/TS 18053 and positions the system within the broader European framework for harmonised digital evidence management.

Alignment with European standardisation is achieved primarily through the adoption of the Custody Transfer Point (CTP) model and the structured Digital Custody Metadata (DCM) approach. By implementing these concepts as core architectural elements rather than as descriptive references, the system ensures that its custody lifecycle, resource attribution model and metadata governance processes are compatible with the terminology and data structures defined at European level. This structural compatibility facilitates future integration with other systems developed under the same or related standards.

Interoperability is further supported through the use of open and widely recognised technical specifications. The system exposes a RESTful API described using OpenAPI/Swagger definitions, enabling machine-readable interface documentation and automated client integration. Data exchange is conducted using structured JSON formats over secure HTTPS communication channels, avoiding proprietary protocols or closed data schemas. This design allows third-party systems, national forensic platforms or cross-border coordination tools to interface with the custody system without architectural redesign.

The backend implementation, based on standardised database technologies and widely adopted authentication mechanisms, further enhances portability and scalability across different operational environments. Role definitions and custody workflows are defined in generic, mission-oriented terms rather than jurisdiction-specific structures, ensuring that the system can be adapted to varying national procedural contexts while maintaining consistent digital custody governance principles.

By combining conceptual alignment with CEN/TS 18053 and adherence to open technical standards, the system contributes to the harmonisation of digital chain-of-custody practices across Member

States. It provides a foundation for cross-border cooperation, supports potential integration within European crisis response mechanisms and facilitates future standardisation developments in the domain of CBRN digital evidence management.

3.3 Roles In Standard / App

3.3.1 Role Model Defined in CEN/TS 18053

CEN/TS 18053 defines a structured stakeholder model within the digital Chain of Custody (dCoC) process. The standard identifies mission-assigned resources operating under defined responsibilities, with custody ownership explicitly attributed at each Custody Transfer Point (CTP). The principal operational roles typically include the Mission Command Team, Reconnaissance Team, Sampling Team, Carrier Team and Laboratory Team.

Within this framework, custody is transferred between authorised resources acting in clearly defined roles. At each CTP, a custody owner and a custody receiver must be identifiable, and their responsibilities must be traceable within the digital custody metadata. The role model is therefore not organisational in a hierarchical sense, but functional: it defines who may collect, transport, receive, analyse or otherwise handle digital evidence items, and ensures that responsibility is continuously attributable throughout the custody lifecycle.

This conceptual separation of roles is fundamental to ensuring traceability, accountability and compliance with evidentiary governance principles in CBRN contexts.

3.3.2 Role Model Implemented in the CBRN Sample Tracking System

The CBRN Sample Tracking System implements a role model derived from the functional structure described in CEN/TS 18053, translated into operational application roles. In the current system iteration, three primary roles are implemented: Documenter, Transporter and Laboratory.

The Documenter role corresponds functionally to the Sampling Team defined in the standard. This role is responsible for registering samples, assigning identifiers and NFC tags, associating samples with collections and initiating custody transfers.

The Transporter role reflects the Carrier Team concept. This role is responsible for receiving custody of collections, confirming transfers and initiating subsequent handovers within the custody chain.

The Laboratory role corresponds to the Laboratory Team defined in the standard. This role is responsible for confirming receipt of samples or collections and recording subsequent custody-related actions within the analytical phase.

Roles are enforced through role-based access control mechanisms at both application and backend service level. Access to system functions and custody operations is restricted according to the authenticated role, ensuring that users can only perform actions consistent with their assigned operational responsibility.

3.3.3 Role Responsibilities Within the Custody Transfer Lifecycle

Within the implemented custody lifecycle, each role is associated with clearly defined operational capabilities. The Documenter may create and document samples, assign digital identifiers, associate

samples with collections and initiate custody transfer events. The Transporter may confirm receipt of a collection, thereby assuming custody, and may initiate onward transfers as required. The Laboratory may confirm receipt and record laboratory-level custody status updates.

Custody transfers require explicit confirmation by authenticated roles, ensuring that both transfer and reception are attributable to identified resources. No single role is capable of completing the entire lifecycle of a custody chain independently. This separation of duties reflects the CTP model described in CEN/TS 18053 and supports accountability through structured role interaction.

By mapping conceptual stakeholder roles from the European standard to clearly defined application roles, the system ensures that digital custody governance remains consistent with recognised operational models while being enforceable through technical controls within the application architecture.

3.4 Custody Transfer Point (CTP) Functionality

The Custody Transfer Point (CTP) constitutes the central operational mechanism of the CBRN Sample Tracking System and represents the moment at which custodianship of a sample or collection is formally transferred from one authorised resource to another. The CTP is implemented as a discrete, verifiable and time-stamped event within the digital Chain of Custody (dCoC), ensuring that each transition of responsibility is explicitly recorded and attributable.

In accordance with the principles defined in CEN/TS 18053, each CTP involves two authenticated resources: a custody owner and a custody receiver. The custody owner initiates the transfer of a sample or collection, while the custody receiver confirms acceptance. The transfer is only considered complete once the receiving resource has formally acknowledged the transfer within the system. This dual confirmation mechanism ensures that responsibility cannot be implicitly assumed and that accountability is continuously maintained.

Within the application workflow, a CTP may occur at several stages of the lifecycle, including transfer from field collection to transport, from transport to laboratory, or between intermediate operational units. Each transfer is associated with structured Digital Custody Metadata capturing mission identifiers, sample or collection identifiers, role attribution, timestamps and transfer status. This metadata forms part of the immutable event log and allows full reconstruction of the custody chain.

The CTP functionality is implemented using an event-based architecture. Each confirmed transfer generates a new custody event recorded within the append-only event store. Rather than modifying previous records, the system records the transfer as an additional chronological event, preserving the integrity of the historical custody trail. The current custody state displayed in the operational interface is derived from the underlying event sequence, ensuring consistency between operational use and evidentiary record.

Validation mechanisms are embedded within the CTP workflow. Prior to confirming a transfer, the system verifies that required metadata elements are present and that the acting resource is authorised to perform the operation. Where discrepancies arise, such as incomplete metadata or role mismatches, the system prevents confirmation of the transfer while recording the attempted action. This ensures transparency without permitting silent alteration or bypass of governance controls.

The system also accommodates operational constraints, including low-connectivity environments. When a transfer is initiated in offline conditions, the custody event is locally recorded and queued for synchronisation. Upon restoration of connectivity, the event is reconciled with the central event store in chronological order. This mechanism preserves continuity of custody while ensuring that the authoritative record remains complete and coherent.

Through this structured CTP functionality, the system operationalises the conceptual custody transfer model defined in European digital chain-of-custody standards. Each transition of custodianship is explicit, verifiable and traceable, thereby reinforcing evidentiary robustness and supporting subsequent administrative, investigative or judicial review.

4 APPLICATION OPERATIONAL WORKFLOW

4.1 Sample Collection Process

The Sample Collection Process is executed by users assigned the Documenter role within the system. This role corresponds to operational personnel responsible for the physical collection and digital registration of samples during an incident. The workflow ensures that all samples are registered within a defined incident context, enriched with structured metadata and digitally linked to their corresponding physical identifiers.

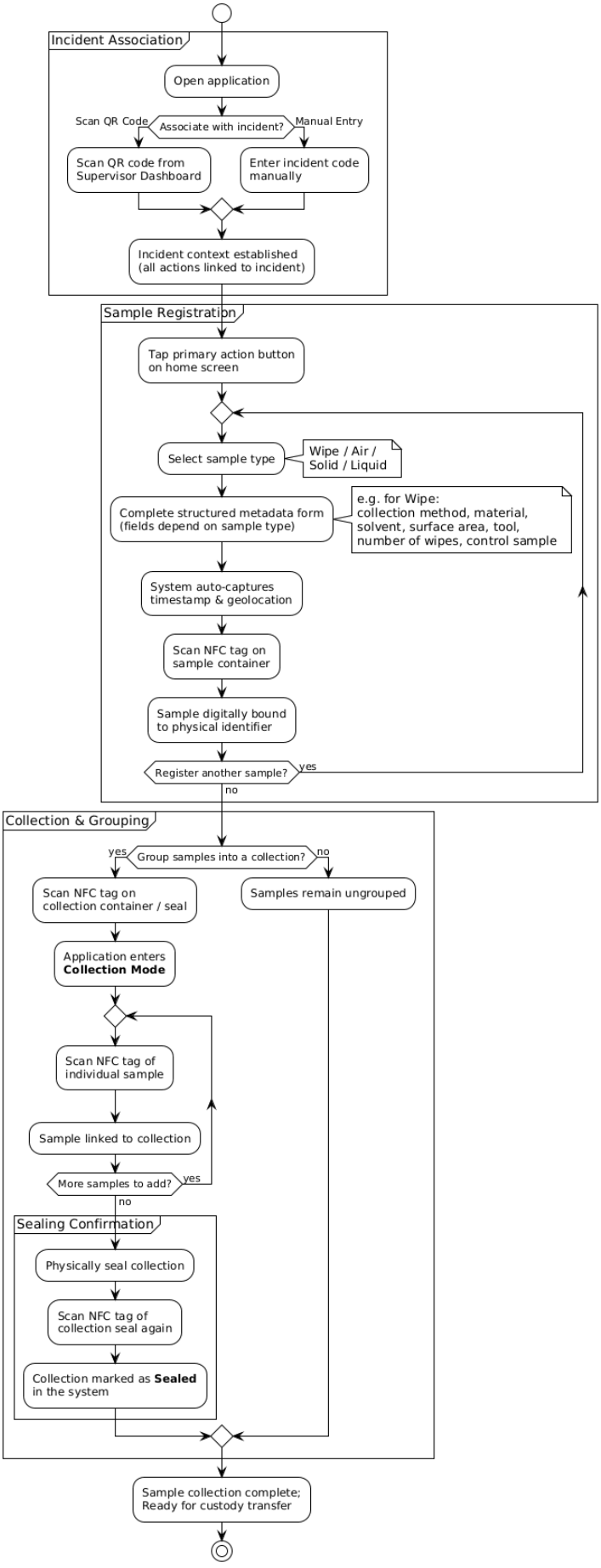

Incident Association
Open application
Associate with incident?
Scan QR Code
Manual Entry
Scan QR code from Supervisor Dashboard
Enter incident code manually
Incident context established (all actions linked to incident)
Sample Registration
Tap primary action button on home screen
Select sample type
Wipe / Air / Solid / Liquid
Complete structured metadata form (fields depend on sample type)
e.g. for Wipe: collection method, material, solvent, surface area, tool, number of wipes, control sample
System auto-captures timestamp & geolocation
Scan NFC tag on sample container
Sample digitally bound to physical identifier
Register another sample?
yes
no
Collection & Grouping
Group samples into a collection?
yes
no
Scan NFC tag on collection container / seal
Samples remain ungrouped
Application enters **Collection Mode**
Scan NFC tag of individual sample
Sample linked to collection
More samples to add?
yes
no
Sealing Confirmation
Physically seal collection
Scan NFC tag of collection seal again
Collection marked as **Sealed** in the system
Sample collection complete; Ready for custody transfer

Figure 1. Activity diagram of sample collection (find larger version in Annex D. Enlarged Sample Collection Activity Diagram)

4.1.1 Incident Association

Before any sampling activity can commence, the Documenter must associate with an active incident. This is achieved by scanning a QR code generated by the Supervisor Dashboard or by manually entering the incident code within the application.

This step establishes the operational context for all subsequent actions. Once signed in, every registered sample, collection and custody event is automatically linked to the selected incident. This ensures that sampling activities remain traceable to a defined geographic and operational framework and prevents the creation of un-associated or orphaned sample records.

4.1.2 Sample Type Selection

After incident association, the Documenter initiates sample registration through the primary action button on the application home screen. The first step in the workflow requires the user to specify the sample type. The current system iteration supports four principal categories:

- Wipe – full Documenter workflow implemented
- Air – partial Documenter workflow implemented
- Solid – sample type selectable, Documenter workflow will be implemented
- Liquid – sample type selectable, Documenter workflow will be implemented

The selected sample type determines the metadata structure and form logic presented to the user. This approach enables the system to enforce structured documentation aligned with the characteristics of each sampling method.

4.1.3 Structured Metadata Capture

Following sample type selection, the user is guided through a structured attribute workflow. The form dynamically presents predefined metadata fields relevant to the selected sampling method. While the available attributes may evolve over time, the system currently supports detailed parameter capture for wipe samples, including:

- Collection method (e.g. wipe or swab)
- Material used
- Solvent applied
- Surface area sampled
- Surface type
- Tool used
- Number of wipes or swabs collected
- Control sample indication

The workflow is implemented as a sequential decision structure, where the selection of one parameter determines the presentation of subsequent fields. This ensures logical consistency and reduces documentation ambiguity.

During the completion of the form, contextual metadata is automatically recorded. This includes timestamp and geolocation data associated with the sampling event. By combining structured user input with automatically captured metadata, the system produces a comprehensive digital representation of the sampling action.

4.1.4 Physical-Digital Binding via NFC

Once the metadata flow is completed, the Documenter assigns a physical identifier to the sample by scanning the NFC tag attached to the sample container. This step creates a binding between the physical evidence item and its digital record within the system.

The NFC identifier becomes the unique reference for the sample throughout its lifecycle. From this point onward, all custody and collection operations reference this identifier,

After successful NFC assignment, the sample registration process is complete. The Documenter may then proceed to register additional samples or group existing samples into a collection. ensuring continuity between the physical and digital domains.

4.1.5 Collection Creation and Grouping

To group multiple samples into a collection, the Documenter scans a new NFC tag associated with the collection container or seal. Upon scanning this identifier, the application enters "collection mode".

In collection mode, the user scans the NFC tags of individual samples to associate them with the collection. Each scan links the sample identifier to the collection record, explicitly defining which items are physically grouped together. This digital grouping mirrors the physical packaging process.

The system allows one or more samples to be added to the collection. The collection record remains editable during this stage, permitting sequential addition of samples as required by operational conditions.

4.1.6 Sealing Confirmation

Once all intended samples have been added, the collection is physically sealed. To reflect this action digitally, the Documenter scans the NFC tag of the collection seal once more. This confirmation scan marks the collection as sealed within the system.

The sealing confirmation represents a significant procedural milestone. After sealing, the collection is considered secured and ready for potential custody transfer. Any subsequent custody movement must follow the defined Custody Transfer Procedure described in Section 4.3.

4.2 Sample Analysis Process

The Sample Analysis Process describes how analytical results generated by external devices are integrated into the CBRN Sample Tracking System. While sampling and custody management are performed through the mobile application and dashboard, analytical measurements may be executed by specialised detection or laboratory devices operating independently of the field application.

The system therefore provides a structured mechanism through which external analysis devices can register analytical activities and associated results within the digital Chain of Custody environment.

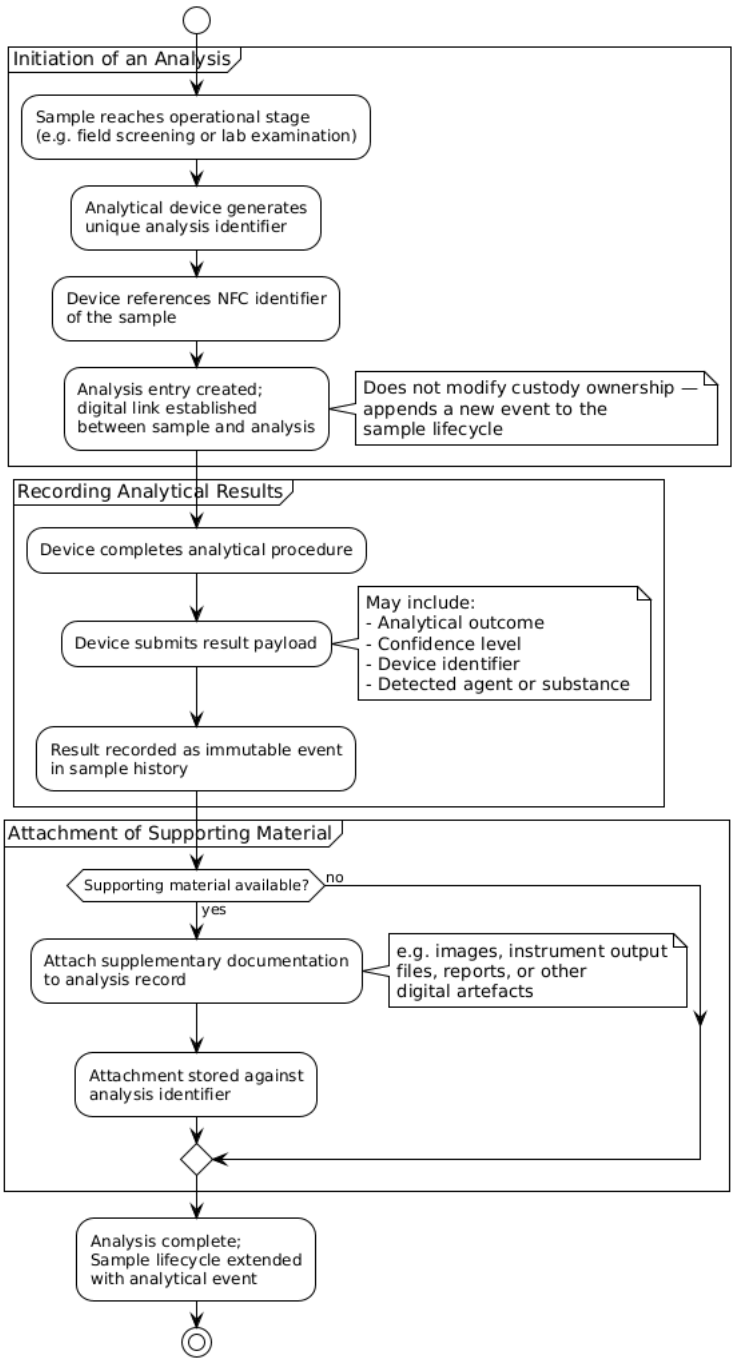


Figure 2. Activity diagram of sample analysis

4.2.1 Initiation of an Analysis

An analysis is initiated by an authorised analytical device once a sample has reached the relevant operational stage (e.g. field screening or laboratory examination).

Each analysis is uniquely identified by an analysis identifier generated by the device performing the measurement. This ensures that the analytical process remains attributable to a specific technical instrument or system and avoids ambiguity where multiple devices may analyse the same sample.

To initiate an analysis, the device references the NFC identifier of the sample. This establishes a direct digital link between the physical sample and the analytical activity. The creation of an analysis entry

does not modify custody ownership but appends a new event to the lifecycle of the sample, thereby extending its evidentiary record.

This design ensures that analytical actions are treated as traceable events within the system rather than as modifications of existing sample records.

4.2.2 Recording Analytical Results

After the analytical procedure has been completed, the device submits the results associated with the previously created analysis record. The system supports a generic result structure designed to accommodate various analytical technologies and methodologies. This setup allows for additional integrations with other analysis devices than the ones provided in the project.

The result payload may include, but is not limited to:

- Analytical outcome (e.g. positive, negative or other classification)
- Confidence level or quantitative certainty indicator
- Device identifier
- Detected agent or substance

By adopting a generic and extensible result schema, the system remains device-agnostic and adaptable to future analytical technologies. This approach allows integration of rapid field detection tools as well as laboratory-based confirmation methods without requiring structural modification of the custody architecture.

The analytical result becomes part of the immutable event history associated with the sample. As such, it is chronologically preserved alongside collection, transfer and verification events.

4.2.3 Attachment of Supporting Material

Where applicable, analytical devices or laboratory systems may attach supplementary documentation to an analysis record. This includes images, instrument output files, reports or other digital artefacts supporting the measurement outcome.

Attachments are stored in association with the specific analysis identifier rather than directly with the sample. This separation ensures that each analytical event retains its own evidentiary documentation and allows multiple analyses to be associated with a single sample without conflict.

The inclusion of attachments strengthens evidentiary robustness by preserving not only the interpreted result but also the underlying supporting material.

4.2.4 Evidentiary Integrity and Traceability

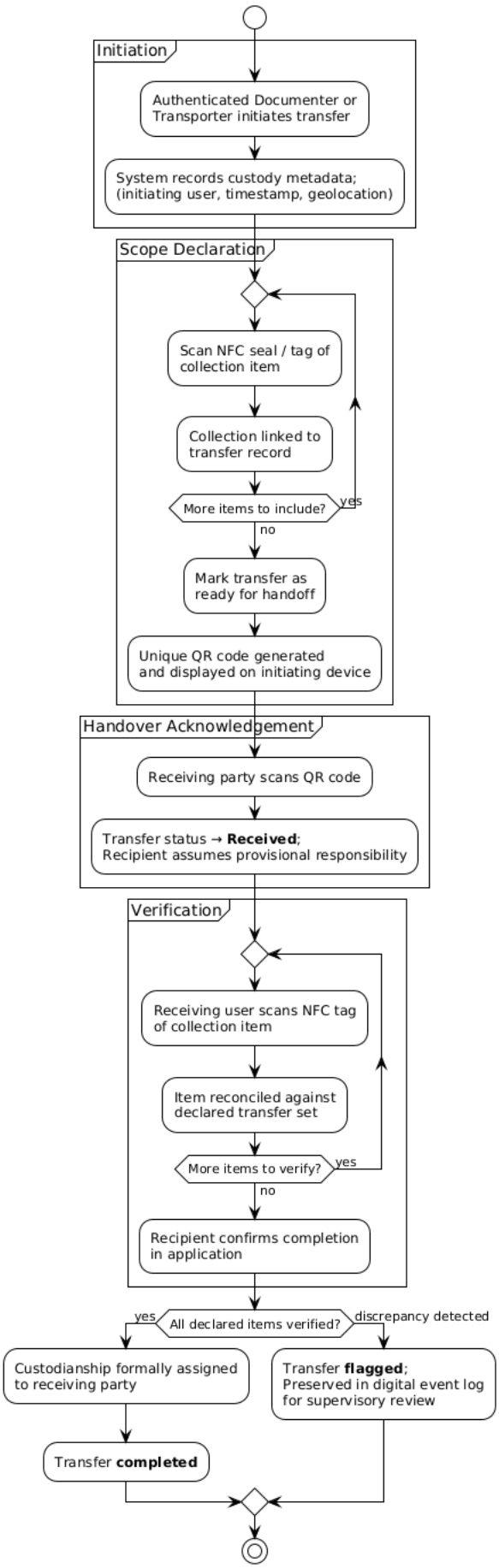
Analytical events are incorporated into the system using the same architectural principles applied to custody operations. Each analysis creation and result submission is recorded as a discrete, time-stamped event attributable to a specific device or authorised analytical resource.

Because the system employs an event-based architecture, analytical activities do not overwrite or alter previous records. Instead, they extend the lifecycle of the sample within the digital log. This ensures that the full progression of a sample—from collection, through custody transfers, to analytical examination—can be reconstructed in chronological order.

The Sample Analysis Process therefore integrates external analytical capabilities into the broader digital Chain of Custody framework while maintaining traceability, device attribution and evidentiary integrity in accordance with European governance principles.

4.3 Custody Transfer Procedure

Custody transfer within the CBRN Sample Tracking System is implemented as a structured, two-party procedure designed to ensure that custodianship of collections is transferred in a verifiable, attributable and auditable manner. The procedure operationalises the Custody Transfer Point (CTP) principle by requiring explicit initiation by an authorised resource and explicit acknowledgement and verification by the receiving resource, thereby maintaining continuous accountability across the transfer lifecycle.


Initiation
Authenticated Documenter or Transporter initiates transfer
System records custody metadata; (initiating user, timestamp, geolocation)
Scope Declaration
Scan NFC seal / tag of collection item
Collection linked to transfer record
More items to include?
yes
no
Mark transfer as ready for handoff
Unique QR code generated and displayed on initiating device
Handover Acknowledgement
Receiving party scans QR code
Transfer status → **Received**; Recipient assumes provisional responsibility
Verification
Receiving user scans NFC tag of collection item
Item reconciled against declared transfer set
More items to verify?
yes
no
Recipient confirms completion in application
yes
All declared items verified?
discrepancy detected
Custodianship formally assigned to receiving party
Transfer **flagged**; Preserved in digital event log for supervisory review
Transfer **completed**

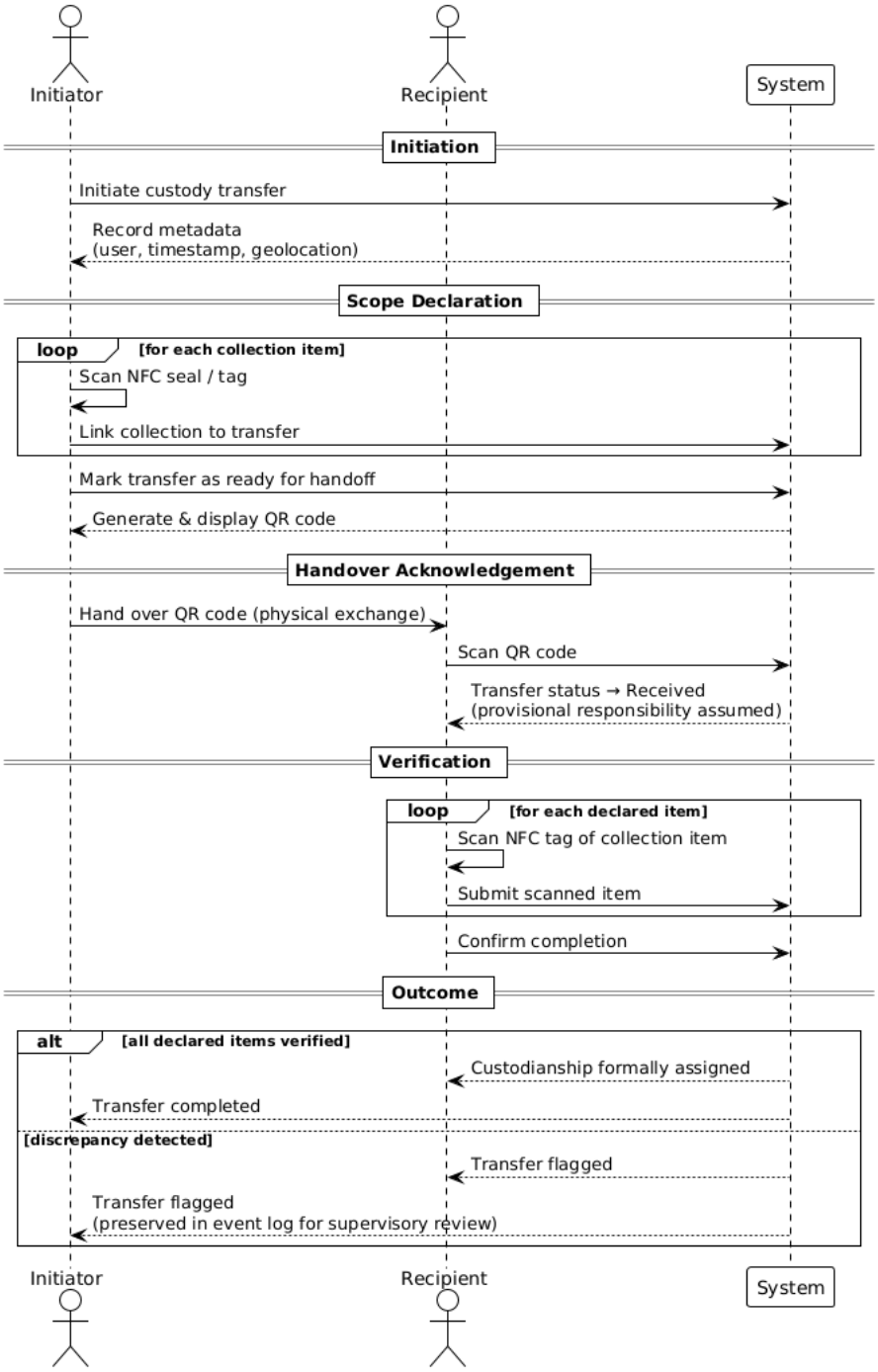


Figure 3. Activity (left) and sequence (right) diagrams for CTP

Currently, a transfer may be initiated by an authenticated Documenter or Transporter, depending on the operational context. Upon initiation, the system automatically records core custody metadata, including the identity of the initiating user, the timestamp and the geolocation of the transfer. This creates a formal digital record of the intended custody transition and establishes the evidentiary starting point of the transfer event.

Following initiation, the initiating user must explicitly define the scope of the transfer. This is performed by scanning the NFC seal or tag of each collection item to be included. Each successful

scan links the corresponding collection identifier to the transfer record, ensuring that the transfer consists only of explicitly declared and digitally bound items. This step prevents ambiguity regarding which collections are subject to custodial change.

Once all intended items have been scanned and associated with the transfer, the initiating user marks the transfer as ready for handoff within the application. At this stage, a unique QR code representing the pending transfer is generated and displayed on the initiating device. The receiving party must scan this QR code to acknowledge the handover. This mechanism ensures that the physical exchange of custody is synchronised with a corresponding digital acknowledgement between two authenticated resources.

When the receiving party scans the QR code, the transfer status changes to “received”. This status confirms that the recipient has acknowledged the transfer request and assumed provisional responsibility for the declared items. However, the transfer is not yet finalised. To ensure completeness and integrity, a verification phase is required.

During verification, the receiving user must scan the NFC tag of each collection item included in the transfer. This item-level reconciliation confirms that the physical items presented correspond exactly to the digitally declared transfer set. Only after all items have been scanned and reconciled may the recipient confirm completion of the transfer within the application.

The system then evaluates whether all expected items were successfully verified. If the declared and verified item sets correspond fully, custodianship is formally assigned to the receiving party, and the transfer is completed. If discrepancies are detected—such as missing, additional or mismatched identifiers—the transfer is flagged accordingly and preserved within the digital event log for supervisory review.

5 TECHNICAL IMPLEMENTATION

5.1 System Architecture and Technology Stack

The CBRN Sample Tracking System is composed of three independently developed and deployed components, each implemented using a technology stack appropriate to its operational context. All three components are written in TypeScript, ensuring a consistent language across the full system and enabling shared type definitions derived from the API contract. The following subsections describe the runtime environment and principal dependencies of each component.

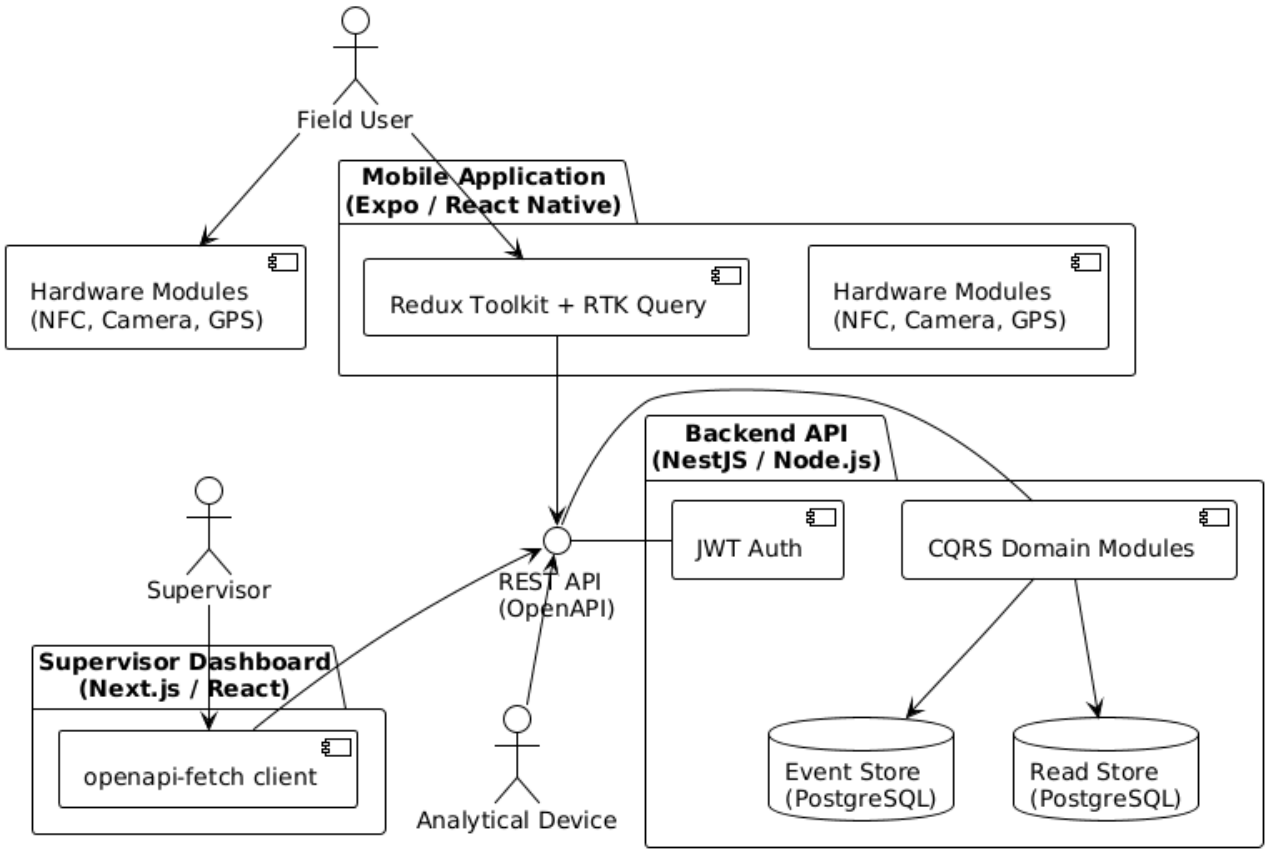


Figure 4. Component diagram

5.1.1 Mobile Application

The mobile field application is built using Expo (SDK 54) and React Native (0.81), targeting both Android and iOS platforms from a single shared codebase. Navigation within the application is managed through Expo Router, which provides a file-system-based routing model consistent with established conventions in modern React frameworks.

Client-side state management and communication with the backend API are handled through Redux Toolkit in combination with RTK Query. The API client layer is automatically generated from the backend's OpenAPI specification using the RTK Query OpenAPI code generator. This approach ensures that all request and response types are derived directly from the API contract, eliminating manual type definitions and reducing the risk of interface inconsistencies between the mobile client and the backend.

Hardware interaction is supported through dedicated Expo and React Native modules. NFC tag scanning is enabled through the react-native-nfc-manager library, which facilitates the binding of physical sample containers and seals to their corresponding digital records. Camera access is provided through expo-camera, enabling QR code scanning during custody transfer operations. Geographic positioning is obtained through expo-location, allowing the recording of location metadata at the point of sample collection or custody transition. Network state detection is provided through expo-network, which enables the application to monitor connectivity status and support operational continuity in degraded network environments.

The user interface is implemented using React Native Paper as the primary component library, supplemented by React Native Reanimated for gesture-driven animations and transitions. Visual elements such as QR code rendering are supported through react-native-qrcode-svg and react-native-svg.

5.1.2 Supervisor Dashboard

The Supervisor Dashboard is a web-based application implemented using Next.js (version 16) and React 19. It follows a server-first rendering model, leveraging Next.js server components and server actions for data retrieval and mutation operations.

Type-safe communication with the backend API is achieved through the combination of openapi-typescript and openapi-fetch. TypeScript type definitions are generated directly from the backend OpenAPI specification, and all API calls are executed through a typed client that enforces conformance to the defined interface contract. Server-side mutation logic is encapsulated using next-safe-action, which provides a structured approach to defining validated server actions with integrated error handling and input schema enforcement via Zod.

Internationalisation is supported through next-intl, enabling locale-aware content rendering and routing. Form handling is managed through react-hook-form in combination with Zod-based validation schemas, ensuring consistent input validation across client and server boundaries.

The user interface is composed of accessible, composable components based on Radix UI primitives, assembled following the shadcn/ui component model. Styling is managed through Tailwind CSS (version 4). Tabular data presentation is handled by TanStack Table, and geographic visualisation of incident locations is provided through Mapbox GL JS, including geocoding-based search functionality via the Mapbox Search JS SDK.

5.1.3 Backend API

The backend API is implemented using NestJS (version 11), running on Node.js 20. It serves as the authoritative data layer and sole point of interaction for both the mobile application and the Supervisor Dashboard.

The application follows a modular, domain-driven architecture organised into bounded context modules — including Incidents, Samples, Collections, Transfers, Analysis and Users — alongside a shared infrastructure module and a dedicated Identity and Access Management (IAM) module. Each context encapsulates its own domain logic, application services, infrastructure components and API presenters in a consistent four-layer structure.

The architectural pattern is based on Command Query Responsibility Segregation (CQRS), implemented through the @nestjs/cqrs module. Write operations are dispatched as commands through a CommandBus and processed by dedicated command handlers that interact with domain aggregates. Read operations are dispatched as queries through a QueryBus and resolved against denormalised read models. Domain events are published through an EventBus and consumed by event handlers responsible for updating read model projections.

Persistence is managed through TypeORM, configured to connect to two separate PostgreSQL databases. The first database serves as the event store, persisting immutable domain events as the authoritative record of all system state changes. The second database functions as the operational read store, containing projected read model entities derived from the event stream. This separation supports independent scaling and preserves evidentiary integrity at the source level.

Authentication is enforced through JSON Web Tokens (JWT), implemented via Passport and the @nestjs/jwt module. All protected endpoints are guarded by a global JWT authentication guard, and role-based access control is applied through a dedicated roles guard. Password hashing is performed using Argon2 to ensure secure credential storage.

The API contract is documented and exposed through Swagger, generated automatically by @nestjs/swagger based on controller metadata and DTO decorators. This specification serves as the single source of truth from which both the mobile application and the Supervisor Dashboard derive their typed API clients. Input validation across all endpoints is handled through class-validator and class-transformer, applied globally via a NestJS validation pipe.

The backend is containerised using Docker. The production Docker Compose configuration orchestrates the API service alongside both PostgreSQL instances, with health checks ensuring that the database services are fully available before the API container starts. The container image is published to the GitHub Container Registry for deployment.

5.2 Data Architecture and Core Entities

The CBRN Sample Tracking System represents its operational domain through a set of core entities that together model the full lifecycle of an incident, from initial response through sample collection, packaging, custody transfer and analysis. Each entity is implemented as a domain aggregate within the backend API, governed by event sourcing principles, and projected into relational read models within the operational database. This section describes the principal entities, their relationships, the identification mechanisms used and the metadata captured to support traceability.

5.2.1 Core Domain Entities

The system is structured around five primary domain entities, each encapsulating a distinct concern within the forensic custody lifecycle.

The **Incident** entity represents the top-level organisational unit to which all operational activities are bound. An incident captures a descriptive name, geographic coordinates (latitude and longitude) indicating the location of the event, and a creation timestamp. An incident may be concluded by an authorised user, at which point the identity of the concluding user and the conclusion timestamp are

recorded. All samples, collections and custody events are associated with a specific incident, ensuring that evidentiary activities are always traceable to a defined operational context.

The **Sample** entity represents an individual specimen collected during field operations. Each sample is classified by type, corresponding to one of four material categories: wipe, air, liquid or solid. A sample is associated with a specific incident and records the identity of the collecting user as well as the timestamp of collection. Additionally, samples support a hierarchical structure through a parent reference, enabling the representation of sub-samples or aliquots derived from an original specimen. Structured metadata may be captured for each sample through a configurable attribute system. Sample attribute definitions specify the metadata fields applicable to a given sample type, including the input type and any predefined options. Individual attribute values are then recorded against each sample, supporting flexible and type-specific documentation of forensic parameters such as solvent information, control sample designation or surface area.

The **Collection** entity represents a sealed grouping of one or more samples. A collection is created by an authorised user and identified by a unique NFC tag corresponding to the physical seal. Once all constituent samples have been added, the collection is sealed, recording both the sealing user and the timestamp at which sealing occurred. The sealing operation is irreversible and marks the collection as ready for custody transfer. A collection maintains a one-to-many relationship with its constituent samples.

The **Transfer** entity models a custody transition between two authorised parties. A transfer records the geographic coordinates of the handoff location, the identity of the initiating user and the initiation timestamp. Transfers contain one or more transfer items, each referencing either an individual sample or a collection. A transfer progresses through a defined status lifecycle consisting of four states: draft, ready, received and verified. During the draft state, items may be added to the transfer. Once handed off, the transfer transitions to a ready state. Upon receipt by the receiving party, the status advances to received, at which point individual items are verified by scanning their physical identifiers. When all items have been verified, the transfer may be completed, transitioning to the verified state. Each item verification records the identity of the verifying user and the verification timestamp. The status transitions are enforced through domain invariants to prevent invalid state changes.

The **Analysis** entity represents an analytical examination performed on a specific sample. An analysis is initiated by referencing the NFC tag of the target sample, establishing a link between the analytical record and the physical specimen. Upon completion, the analysis records the detection device used, a confidence level expressed as a numeric value between zero and one hundred, the analytical result (positive, negative, inconclusive, cancelled or error) and, where applicable, the CAS number of the detected substance. An analysis may additionally contain one or more file attachments, each capturing the original filename, MIME type, storage path and file size, enabling the association of supplementary analytical documentation with the digital record.

5.2.2 Entity Relationships

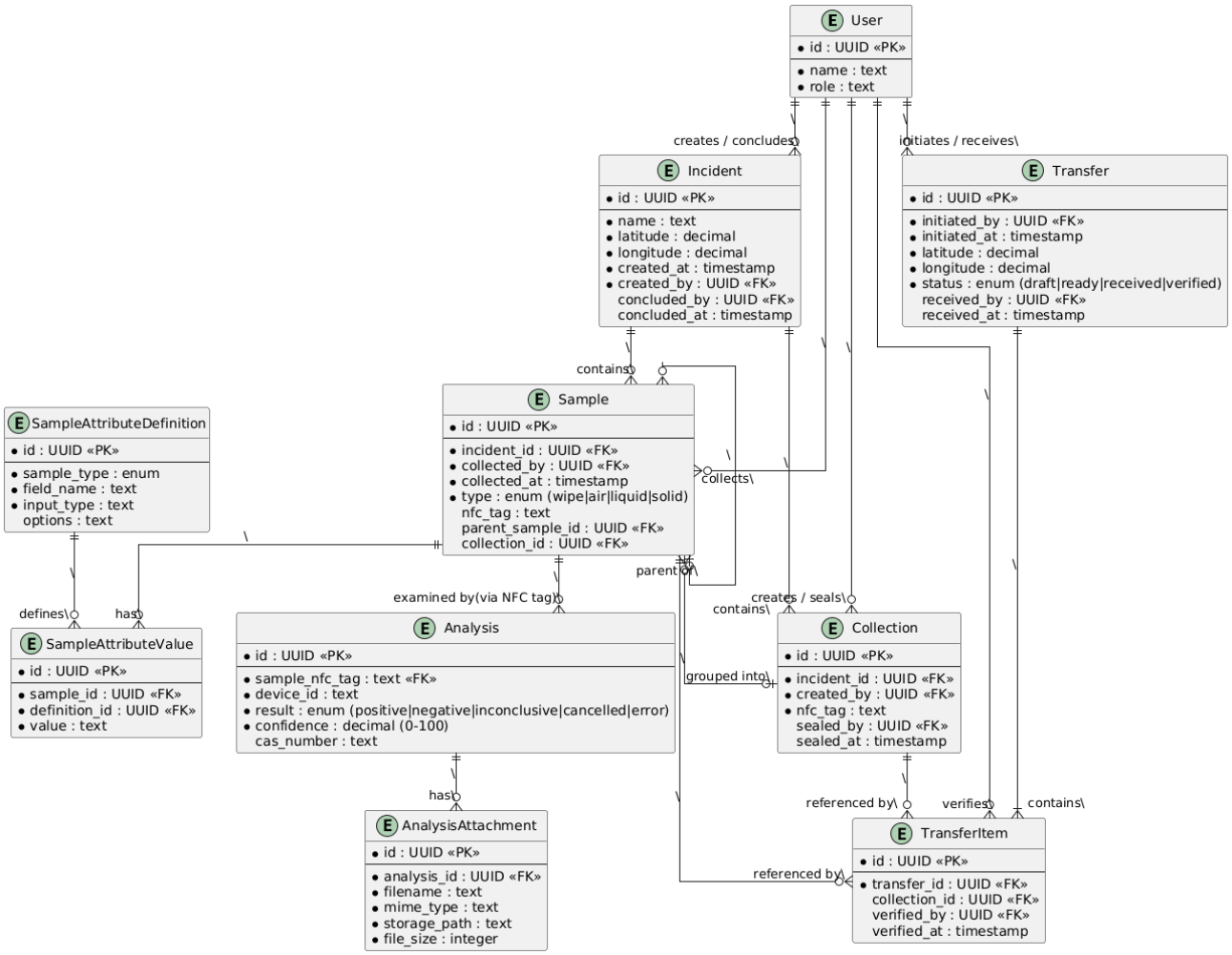


Figure 5. Entity relationship diagram

The relationships between the core entities form a hierarchical structure that reflects the operational progression from incident declaration through to analytical evaluation.

An incident serves as the root entity and maintains a one-to-many relationship with samples. Each sample is associated with exactly one incident, established at the point of collection. Samples are optionally associated with a collection through a many-to-one relationship; a sample may exist independently or may be assigned to a collection prior to sealing. A collection in turn maintains a one-to-many relationship with its constituent samples.

Transfers reference collections and individual samples through their transfer items. Each transfer item holds a reference to either a collection or a sample, enabling the system to represent custody transitions at both the individual specimen level and the sealed container level. An analysis is linked to a sample through the NFC tag, associating analytical results with the physical specimen identified during the analysis process.

Users participate across all entities through foreign key relationships. Each action — whether sample collection, collection creation, collection sealing, transfer initiation, transfer receipt, item verification or incident conclusion — records the identity of the responsible user, ensuring that every operation within the system is attributable to an authenticated individual.

5.2.3 Identification Mechanisms

The system employs multiple identification mechanisms to ensure unambiguous referencing of both digital records and physical objects.

All primary domain entities — incidents, samples, collections, transfers and analyses — are assigned a Universally Unique Identifier (UUID) at the time of creation. These identifiers serve as primary keys within both the event store and the operational read database, ensuring global uniqueness and enabling deterministic referencing across system boundaries.

Physical-digital binding is achieved through NFC identifiers. Each physical sample container and each collection seal may be equipped with an NFC tag. When scanned through the mobile application, the NFC identifier is recorded against the corresponding digital record, establishing a verifiable link between the physical evidence item and its digital representation. The system supports lookup of both samples and collections by their NFC tag, enabling field personnel to retrieve the digital record of any physical item through a single scan operation.

Incident codes provide an additional identification mechanism at the operational level. Supervisors may generate a QR code representation of an incident identifier within the Supervisor Dashboard, enabling field personnel to associate themselves with the correct operational context by scanning the code through the mobile application.

Analysis records are linked to their target sample through the NFC tag of the specimen, establishing a cross-reference between the analytical context and the physical sample without requiring direct knowledge of the sample's internal UUID.

5.2.4 Traceability Metadata

Each entity within the system captures structured metadata to support full traceability, auditability and the reconstruction of operational history.

Temporal metadata is recorded at every significant state transition. Samples record the timestamp of collection. Collections record the timestamp of sealing. Transfers record the timestamps of initiation, receipt and item-level verification. Incidents record both the creation timestamp and, where applicable, the conclusion timestamp. All timestamps are persisted in time zone-aware format to ensure unambiguous temporal ordering regardless of the geographic location of the operation.

Geographic metadata is captured where operationally relevant. Incidents record the geographic coordinates of the event location. Transfers record the geographic coordinates of the custody handoff, enabling the spatial context of custody transitions to be preserved and audited.

User attribution is enforced throughout the system. Every state-changing operation records the identity of the authenticated user who performed the action. Samples record the collecting user. Collections record both the creating and sealing users. Transfers record the initiating user, the receiving user and the user who verified each individual item. Incidents record the user who concluded

the incident. This attribution model ensures that no custody-relevant action occurs without a traceable link to a responsible individual, supporting the principles of non-repudiation and accountability.

5.2.5 Lifecycle Reconstruction

The combination of entities, relationships, identifiers and metadata described above enables the complete reconstruction of the sample lifecycle from initial collection to analytical evaluation. For any given sample, the system can determine the incident under which it was collected, the identity of the collecting officer and the time and circumstances of collection. Through the collection and transfer entities, the system can trace the full chain of custodianship: which samples were grouped together, when and by whom the collection was sealed, to whom custody was transferred, where the transfer took place, and whether each item was verified upon receipt. Through the analysis entity, the system can associate analytical results and supporting documentation with the original specimen.

Because all state transitions are recorded as immutable domain events within the event store, this lifecycle reconstruction is not limited to the current state of each entity. The event log preserves the complete chronological history of every operation, enabling precise reconstruction of the system state at any point in time. This architecture provides the evidentiary foundation required for defensible chain of custody governance in accordance with European digital forensic standards.

5.3 API and External Device Integration

The CBRN Sample Tracking System exposes a centrally governed REST API through which all client applications and external devices interact with the system. This API constitutes the sole integration surface, ensuring that every operation — whether originating from the mobile field application, the Supervisor Dashboard or an external analytical instrument — passes through a single authoritative layer responsible for authentication, authorisation, validation and event persistence. This section describes the API design, its documentation, the authentication model, the integration pathway for external analytical devices and the data format conventions that support device-agnostic interoperability.

5.3.1 REST API Design

The API follows RESTful design conventions. Resources are organised around the principal domain entities — samples, incidents, collections, transfers, analyses and users — each exposed under a dedicated URL namespace. Standard HTTP methods are used to express intent: `POST` for resource creation and command execution, `GET` for retrieval, `PATCH` for partial updates, and `PUT` for idempotent creation or replacement.

Resource endpoints follow a consistent naming convention. Entity collections are addressed at their root path (e.g. `/samples`, `/incidents`, `/collections`, `/transfers`), while individual entities are addressed by their UUID (e.g. `/samples/{id}`, `/incidents/{id}`). Sub-resources and lifecycle actions are expressed through nested paths, such as `/incidents/{id}/samples` for retrieving samples associated with a given incident, `/collections/{id}/seal` for sealing a collection, or `/transfers/{id}/handoff`, `/transfers/{id}/receive` and `/transfers/{id}/verify` for advancing a transfer through its custody lifecycle. NFC-based lookups are exposed through dedicated

paths such as `/samples/by-nfc/{tag}` and `/collections/by-nfc/{tag}`, enabling field-level retrieval of digital records through physical tag identifiers.

All endpoints enforce input validation globally through a validation pipe, ensuring that incoming request bodies conform to defined structural and type constraints before reaching application logic.

5.3.2 OpenAPI Specification and Documentation

The API contract is formally documented through an OpenAPI specification generated automatically at runtime by the `@nestjs/swagger` module. Controller metadata, Data Transfer Object (DTO) decorators and response type annotations are introspected to produce a machine-readable specification exposed at the `/swagger/json` endpoint. A visual Swagger UI is additionally available at the `/api` path, providing interactive documentation for developers and integrators.

Each endpoint is annotated with an operation identifier, a summary, parameter descriptions and response types, ensuring that the specification is sufficiently detailed to serve as the single source of truth for client development. Both the mobile application and the Supervisor Dashboard derive their typed API client code directly from this specification through automated code generation tooling, as described in Section 5.1.

This approach ensures that the API documentation is always synchronised with the implementation. Any change to a controller, DTO or response model is automatically reflected in the published specification, reducing the risk of documentation drift and enabling integration partners to consume an accurate and up-to-date interface definition.

5.3.3 Authentication Requirements

Access to the API is governed by a JSON Web Token (JWT) authentication model. Clients authenticate by submitting credentials to the `/authentication/sign-in` endpoint, which returns an access token and a refresh token upon successful verification. The access token must be included as a Bearer token in the `Authorization` header of all subsequent requests to protected endpoints. Token refresh is supported through the `/authentication/refresh-tokens` endpoint, enabling continued access without re-authentication.

All endpoints are protected by a global JWT authentication guard by default. Endpoints that must be accessible without authentication — such as the analysis endpoints intended for integration with external devices — are explicitly annotated with the `@Public()` decorator, exempting them from the authentication requirement.

Role-Based Access Control (RBAC) is enforced through a roles guard. Each endpoint declares the roles permitted to invoke it through the `@Roles()` decorator. For example, incident creation and conclusion are restricted to the `Supervisor` role, sample creation is permitted for both `Supervisor` and `Documenter` roles, and custody transfer operations are accessible to `Documenter` and `Transporter` roles. Requests from users whose role does not match the required permissions are rejected.

5.3.4 Integration of External Analytical Devices

A dedicated integration pathway is provided for external analytical devices and laboratory systems through the `/analysis` endpoint namespace. This interface is designed to accommodate any

analytical instrument capable of issuing HTTP requests, regardless of manufacturer, technology or measurement methodology.

The analysis integration follows a three-phase interaction model corresponding to the lifecycle of an analytical examination: registration, result submission and attachment of supporting material.

To register an analysis, the external device issues a `PUT` request to `/analysis/{id}`, providing a device-generated identifier for the analysis and the NFC tag of the target sample in the request body (via the `sampleNfcTag` field). The device-generated identifier allows the initiating instrument to maintain referential control over the analysis record. The NFC tag establishes the link between the analytical activity and the physical specimen. Upon processing, the system creates a new `Analysis` aggregate and records an `AnalysisStartedEvent` within the event store.

Upon completion of the analytical procedure, the device submits results by issuing a `PATCH` request to `/analysis/{id}`. The result payload includes the `confidence_level` (a numeric value between zero and one hundred), a `device` identifier indicating which instrument performed the analysis, the analytical `result` (one of `positive`, `negative`, `inconclusive`, `cancelled` or `error`) and, where a substance has been detected, the CAS number of the identified `agent`. This structure is intentionally generic and extensible, accommodating both rapid field detection tools and laboratory-based confirmation methods without requiring structural modification of the API.

Where supplementary documentation is available, the device may attach files to the analysis record by issuing a `POST` request to `/analysis/{id}/attachment` with a `multipart/form-data` payload. Up to five files may be uploaded in a single request. Each attachment is stored with its original filename, MIME type, a system-generated unique filename and its file size, enabling the preservation of instrument output files, images, reports or other digital artefacts supporting the analytical result.

All three analysis endpoints are currently marked as public, allowing external devices to interact with the system without requiring JWT-based authentication. This design decision reflects the operational constraint that many analytical instruments operate as autonomous devices without an interactive user session. API key-based authentication for these endpoints is identified as a planned enhancement for future iterations.

5.3.5 Data Formats and Communication Conventions

All API communication uses JSON as the default data interchange format for request and response bodies. JSON payloads are structured according to the schemas defined in the OpenAPI specification, with field names following `camelCase` convention. Timestamps are represented in ISO 8601 format with time zone information. Identifiers are represented as UUID strings. Enumerated values — such as sample types, user roles and transfer statuses — are transmitted as lowercase string literals corresponding to their domain definitions (e.g. `"wipe"`, `"documenter"`, `"draft"`).

File uploads represent the sole exception to JSON-based communication. Attachment submission to the analysis endpoint uses `multipart/form-data` encoding, consistent with standard HTTP conventions for binary file transfer.

Response codes follow standard HTTP semantics: `201` for successful resource creation, `200` for successful retrieval and updates, `400` for validation failures, `401` for authentication errors and `500` for

internal server errors. Error responses include a structured body containing a `message`, an `error` classification and the corresponding `statusCode`.

5.3.6 Interoperability and Device-Agnostic Design

The API is designed to be device-agnostic. No assumption is made regarding the manufacturer, operating system or communication capabilities of the integrating device, beyond the ability to issue standard HTTP requests and produce JSON payloads. The analysis integration interface in particular avoids proprietary data structures or device-specific protocols, relying exclusively on open standards (HTTP, JSON, `multipart/form-data`) and generic result schemas.

The OpenAPI specification serves as the formal integration contract, enabling any system capable of consuming an OpenAPI definition to generate client bindings automatically. This supports integration not only with the project's own mobile and web applications but also with third-party laboratory information management systems (LIMS), portable detection instruments and other operational systems within the CBRN response ecosystem.

By adopting a generic and extensible result schema for the analysis interface, the system is prepared to accommodate future analytical technologies and methodologies without requiring structural modification of the API contract. New result fields or attachment types can be introduced through additive schema changes while maintaining backward compatibility with existing integrators.

5.4 Security and Access Control

The CBRN Sample Tracking System enforces a layered security model designed to protect evidentiary data, ensure that only authorised individuals may perform custody-relevant operations, and maintain full accountability for all actions within the system. Security is implemented through authentication at the API boundary, role-based access control at the endpoint level, domain-level invariants that prevent invalid state transitions, and architectural guarantees that preserve the integrity of the evidentiary record. This section describes each of these mechanisms and their contribution to the overall security posture of the system.

5.4.1 Authentication Mechanism

All access to the backend API is governed by a token-based authentication model using JSON Web Tokens (JWT). The Identity and Access Management (IAM) module implements this model through a dedicated authentication service, a Passport-based JWT strategy and a globally registered authentication guard.

To obtain access, a client submits credentials (username and password) to the `/authentication/sign-in` endpoint. The authentication service retrieves the corresponding user record and verifies the submitted password against the stored hash using Argon2, a memory-hard hashing algorithm selected for its resistance to brute-force and side-channel attacks. Upon successful verification, the service issues a token pair consisting of an access token and a refresh token.

The access token is a signed JWT containing the user's identifier (`sub`), username (`name`) and role (`role`). It is configured with a defined issuer, audience and time-to-live (defaulting to 3600 seconds). The token is signed using a server-side secret and must be included as a Bearer token in the

`Authorization` header of all subsequent requests to protected endpoints. The JWT strategy validates incoming tokens by verifying the signature, issuer, audience and expiration, and rejects requests bearing expired or tampered tokens.

The refresh token enables session continuity without requiring re-authentication. When the access token expires, the client may submit the refresh token to the `/authentication/refresh-tokens` endpoint to obtain a new token pair. The system implements refresh token rotation: upon each successful refresh, the previous refresh token is invalidated and a new one is issued. This mechanism limits the window of exposure in the event of token compromise. If an invalidated refresh token is presented — indicating potential token theft — the system raises an `InvalidatedRefreshTokenError` and denies access, providing a detection signal for compromised sessions.

The `JwtAuthGuard` is registered as a global guard through the NestJS dependency injection container, ensuring that authentication is enforced across all endpoints by default. Endpoints that must be accessible without authentication — such as the analysis endpoints intended for external device integration — are explicitly exempted through the `@Public()` decorator, which sets metadata inspected by the guard to bypass token validation for those specific routes.

5.4.2 Role-Based Access Control

Authorisation is enforced through a Role-Based Access Control (RBAC) model. Each user within the system is assigned a single role that defines the scope of operations they are permitted to perform. The system defines five roles, each corresponding to a distinct operational responsibility within the CBRN response and forensic custody lifecycle:

The `Supervisor` role is assigned to personnel responsible for incident governance and operational oversight. Users with this role may create and update incidents, conclude incidents, manage user accounts and view incident-level data including associated samples. Within the Supervisor Dashboard, this role provides full access to administrative and monitoring capabilities.

The `Documenter` role is assigned to field personnel responsible for sample collection and documentation. Users with this role may create samples, assign NFC tags, create and seal collections, initiate custody transfers and retrieve sample and collection data. This role represents the primary operational actor in field environments.

The `Transporter` role is assigned to personnel responsible for the physical transport of evidence between locations. Users with this role may initiate and participate in custody transfers, receive transfers, and verify individual transfer items upon receipt. The verification of transfer items — confirming that physical identifiers match declared digital records — is restricted to this role, ensuring that only the receiving party may acknowledge receipt.

The `Analyzer` role is assigned to personnel or systems responsible for performing analytical examinations on samples. This role corresponds to the laboratory function within the custody lifecycle.

The `Admin` role provides unrestricted access to all endpoints, bypassing role-based restrictions entirely. This role is intended for system administration purposes.

5.4.3 Backend Enforcement of Permissions

Role-based permissions are enforced exclusively at the backend level through the `RolesGuard`, which is registered as a global guard alongside the `JwtAuthGuard`. Each controller endpoint declares its permitted roles through the `@Roles()` decorator. When a request arrives, the guard extracts the authenticated user's role from the JWT payload and evaluates it against the declared role requirements. If no roles are declared for an endpoint, access is permitted to any authenticated user. If roles are declared, the request is permitted only if the user's role matches at least one of the specified values, or if the user holds the `Admin` role.

This enforcement model ensures that authorisation decisions are made centrally and consistently, independent of the client application. Neither the mobile application nor the Supervisor Dashboard implements its own access control logic; both rely entirely on the API to accept or reject operations based on the authenticated user's role. This prevents the possibility of client-side circumvention and ensures that the same access control rules apply regardless of the integration channel through which a request originates.

The following table summarises the role-based permissions enforced across the principal API operations:

Operation	Permitted Roles
Create / update / conclude incident	Supervisor
List / view incidents	Supervisor
View individual incident	Supervisor, Documenter
Create sample	Supervisor, Documenter
List / view / update samples	Documenter
View individual sample	Supervisor, Documenter
Create / seal collection	Documenter
View collection	Documenter, Supervisor
Create / handoff / receive / verify transfer	Documenter, Transporter
Verify individual transfer item	Transporter
Create / list users	Supervisor
Register / complete analysis, attach files	Public (no authentication required)

Table 1. Role-Based Access Control Matrix for Principal API Operations

5.4.4 Protection Against Unauthorised Modification

The system's event-sourced architecture provides a structural guarantee against unauthorised modification of evidentiary data. All state changes are recorded as immutable domain events within an append-only event store. Once an event has been persisted — whether representing sample

creation, collection sealing, custody transfer or analytical result submission — it cannot be altered or deleted through normal system operations. The event store uses a composite key of stream identifier and position number with a unique index constraint, preventing the insertion of duplicate or out-of-sequence events.

Read models, which serve the operational queries of the mobile application and Supervisor Dashboard, are derived projections of the event stream. Even if a read model were to be compromised or corrupted, the authoritative event log remains intact and can be used to reconstruct the correct system state at any point in time.

Domain aggregates enforce additional invariants that prevent invalid state transitions. For example, a collection cannot be sealed more than once, transfer items cannot be verified unless the transfer is in the `Received` state, items cannot be added to a transfer that has already been handed off, and a transfer cannot be marked as verified unless all constituent items have been individually confirmed. These invariants are evaluated within the domain layer before any event is committed, ensuring that the business rules governing evidentiary operations are enforced regardless of the content of incoming requests.

5.4.5 Secure Communication

In production deployments, all communication between client applications and the backend API is conducted over HTTPS, ensuring that data in transit — including authentication credentials, JWT tokens and evidentiary payloads — is protected by TLS encryption. This prevents interception, tampering and replay of sensitive information during transmission. The enforcement of transport-layer security is handled at the infrastructure level through reverse proxy or load balancer configuration, consistent with standard deployment practices for containerised services.

5.4.6 Accountability Through Identity-Linked Actions

Every state-changing operation within the system is attributable to a specific authenticated user. The JWT payload carried with each request contains the user's identifier and role, which are extracted by the `@ActiveUser()` decorator and propagated to the application layer. When a command is executed — such as creating a sample, sealing a collection, initiating a transfer or concluding an incident — the identity of the acting user is recorded as part of the resulting domain event and persisted in both the event store and the operational read model.

This attribution model ensures that the system maintains a complete and tamper-resistant audit trail. For any given entity, it is possible to determine which user created it, which user modified it and at what time each action occurred. In the context of custody transfers, both the initiating user and the receiving user are recorded, along with the user who verified each individual item. In the context of incidents, the user who concluded the incident is recorded alongside the conclusion timestamp.

Because user identity is derived from the server-validated JWT rather than from client-supplied parameters, the attribution cannot be falsified by a compromised or malicious client. The combination of authenticated identity, role-based authorisation, immutable event recording and domain-enforced invariants provides a comprehensive accountability framework consistent with the non-repudiation requirements of European digital Chain of Custody standards.

5.5 Event Logging, Traceability and Synchronisation

The CBRN Sample Tracking System is designed to guarantee full traceability and auditability of every custody-relevant operation. This guarantee is realised through an event-sourced architecture in which all state changes are recorded as immutable events, a strict separation between write and read models through CQRS, and a synchronisation mechanism that enables the mobile application to operate in disconnected field environments while preserving the completeness and chronological integrity of the evidentiary record. This section describes the technical implementation of each of these mechanisms.

5.5.1 Event Sourcing Implementation

All write operations within the system are modelled as domain events that capture the fact of a state change rather than overwriting existing records. When a command is processed — such as creating a sample, sealing a collection or initiating a custody transfer — the responsible command handler interacts with a domain aggregate. The aggregate validates the operation against its current state and its domain invariants, and upon successful validation, produces one or more domain events describing the resulting state transition.

Each domain aggregate extends `VersionedAggregateRoot`, a base class that maintains a monotonically increasing version counter. When events are committed, the `EventStorePublisher` serialises each event through the `EventSerializer`, which records the aggregate's `streamId` (its UUID), the event `position` (derived from the aggregate's current version incremented by one), the event `type` (the class name of the domain event) and the event `data` (a JSON representation of the event payload). For operations that produce multiple events within a single command, the publisher assigns sequential position values to ensure correct ordering.

Domain event classes are registered in a global `EventClsRegistry` through the `@AutoWiredEvent` decorator. This registry enables the `EventDeserializer` to reconstruct typed event instances from their persisted JSON representation when events are retrieved from the store, ensuring that replayed events retain their original class identity and can be processed by the appropriate event handlers.

5.5.2 Append-Only Event Storage

The event store is implemented as a PostgreSQL database containing a single `events` table. Each record in this table comprises a `streamId` (identifying the aggregate to which the event belongs), a `position` (indicating the sequential order of the event within that aggregate's history), the event `type`, the event `data` stored as a JSON column, an optional `userId` and a `createdAt` timestamp recorded in timezone-aware format.

A unique composite index on `(streamId, position)` enforces the append-only guarantee at the database level. This constraint ensures that no two events may occupy the same position within a given stream, preventing duplicate or out-of-sequence insertions. Events are persisted within database transactions managed through a dedicated query runner: if any event in a batch fails to persist, the entire transaction is rolled back, ensuring atomicity.

Once persisted, events are never modified or deleted through normal system operations. This immutability constitutes the foundational integrity guarantee of the system. The event store serves as the authoritative and tamper-resistant record of every action performed within the custody lifecycle.

5.5.3 CQRS Separation of Write and Read Models

The system implements the Command Query Responsibility Segregation (CQRS) pattern, maintaining a strict architectural boundary between write operations and read operations.

Write operations flow through the `CommandBus`, which dispatches commands to dedicated command handlers. Each handler interacts with the relevant domain aggregate — either constructing a new aggregate through a factory or rehydrating an existing aggregate from its event history — applies the requested mutation, and commits the resulting events. The events are persisted to the event store and simultaneously dispatched through the `EventBus`.

A TypeORM `EventEntitySubscriber` listens for insert operations on the event store. After each successful insertion, the subscriber forwards the event data to the NestJS `EventBus`, which in turn dispatches it to registered event handlers. These event handlers — such as `SampleCreatedEventHandler`, `IncidentCreatedEventHandler` or `TransferCreatedEventHandler` — project the event data into denormalised read model entities within a separate operational PostgreSQL database. Each entity type (e.g. `SampleEntity`, `IncidentEntity`, `CollectionEntity`, `TransferEntity`, `AnalysisEntity`) is optimised for the query patterns required by the mobile application and the Supervisor Dashboard.

Read operations flow through the `QueryBus`, which dispatches queries to dedicated query handlers. These handlers retrieve data exclusively from the read model repositories, without interacting with the event store or domain aggregates. This separation ensures that read-heavy operations do not contend with write-path logic and that the read models can be independently optimised, rebuilt or extended without affecting the authoritative event log.

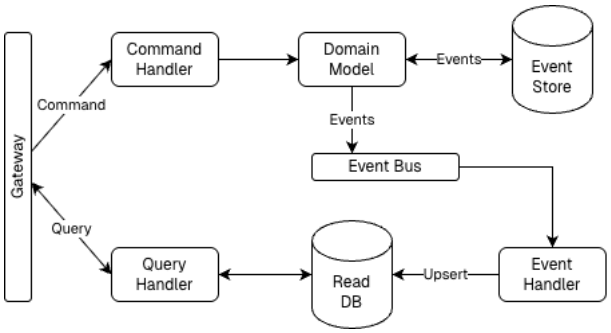


Figure 6. Activity diagram of CQRS and event sourcing architecture

5.5.4 Aggregate Rehydration and Lifecycle Reconstruction

When a write operation targets an existing entity — such as updating an incident, sealing a collection or verifying a transfer item — the system must reconstruct the aggregate's current state before applying the mutation. This is performed by the `AggregateRehydrator`, which retrieves all events for a given `streamId` from the event store, ordered by position in ascending sequence, and replays them against a fresh aggregate instance.

Each event is applied to the aggregate through a handler method named by convention (e.g. `onIncidentCreatedEvent`, `onCollectionSealedEvent`). As each event is replayed, the aggregate's internal state is progressively restored to reflect the cumulative effect of all prior operations. The

aggregate's version is set to the position of the last event in the stream, ensuring that any new events committed by the current operation will be assigned the correct subsequent position.

This rehydration mechanism means that the complete lifecycle of any entity can be reconstructed at any time by replaying its event stream. For a given sample, this includes its creation, NFC tag assignment, collection association, custody transfers, and any associated analytical activities. The event store thereby provides a full chronological history that supports forensic audit, dispute resolution and regulatory compliance.

5.5.5 Offline Operation of the Mobile Application

The mobile field application is designed to operate in environments where network connectivity may be intermittent, unreliable or entirely absent. This is a critical operational requirement for CBRN response scenarios, where field conditions frequently preclude stable network access.

To support offline operation, the mobile application maintains a dual-store architecture within its Redux state management layer. For each entity type that may be created in the field — samples, sample attribute details and collections — the store contains both a local slice and a synced slice. When the application detects that network connectivity is unavailable, newly created entities are persisted to the corresponding local slice within the in-memory Redux store. The application continues to function normally: users may register samples, capture structured metadata, assign NFC tags and create collections without interruption, regardless of connectivity status.

Network state is continuously monitored through the `expo-network` module. A `NetworkProvider` component wraps the entire application and tracks the current connectivity status via the `useNetworkState` hook. Changes in connectivity are dispatched to a global configuration slice within the Redux store, making the connection state available to all components and synchronisation logic throughout the application.

5.5.6 Synchronisation of Locally Stored Events

When network connectivity is restored, the mobile application automatically initiates synchronisation of all locally queued entities with the backend API. This process is managed by a dedicated `SyncEngine` component that operates as a background synchronisation layer within the application's component tree.

The `SyncEngine` monitors three local entity queues: local samples, local sample details and local collections. When connectivity becomes available and one or more queues contain pending items, the engine processes each queue sequentially, submitting each locally stored entity to the corresponding API endpoint using the standard RTK Query mutation hooks (`useCreateSampleMutation`, `useCreateSampleDetailMutation`, `useCreateCollectionMutation`).

Each synchronisation attempt is wrapped in a retry mechanism with exponential backoff. If a submission fails, the engine retries up to three times with progressively increasing delay intervals (500 ms, 1000 ms, 2000 ms). Upon successful submission, the server-confirmed entity is added to the corresponding synced slice and the local copy is removed from the local queue. If all retry attempts are exhausted, the entity remains in the local queue for subsequent synchronisation attempts, and the failure is logged for diagnostic purposes.

The synchronisation process is designed to be cancellable and idempotent. If the user navigates away or connectivity is lost during synchronisation, the process terminates gracefully and resumes from where it left off when conditions permit. UUIDs are generated client-side at the point of entity creation, ensuring that each entity retains a stable and globally unique identifier regardless of when it is synchronised with the backend. This prevents the creation of duplicate records in the event of interrupted or repeated synchronisation attempts.

5.5.7 Handling of Connectivity Interruptions

The system accommodates connectivity interruptions at multiple levels. At the application level, the dual-store architecture ensures that field personnel experience no functional degradation when operating without network access. All sampling, metadata capture and collection workflows remain fully available. The application presents both locally stored and synced entities through unified hooks (e.g. `useCollections`), providing a seamless user experience regardless of the underlying synchronisation state of individual records.

At the synchronisation level, the queue-based approach ensures that no data is lost during connectivity interruptions. Entities created offline are retained in the local Redux store until they have been successfully transmitted to and acknowledged by the backend API. The synchronisation engine activates automatically upon connectivity restoration, requiring no manual intervention from the user.

At the backend level, the event-sourced architecture provides natural resilience against ordering concerns. Each event is assigned its position within its aggregate stream based on the aggregate's version at the time of processing, and timestamps are recorded server-side upon persistence. This ensures that the authoritative chronological ordering within the event store is determined by the backend at the time of receipt rather than by the client at the time of creation, while the client-side timestamps captured in the entity payloads (such as the `collectedAt` timestamp on samples) preserve the actual time of the field operation.

5.5.8 Preservation of Chronological Integrity

The system employs several complementary mechanisms to preserve the chronological integrity of the evidentiary record.

Within the event store, monotonic position numbering within each aggregate stream ensures strict ordering of events belonging to the same entity. The unique composite index on `(streamId, position)` enforces this ordering at the database constraint level, making it structurally impossible to insert events out of sequence.

All timestamps within the event store are recorded using PostgreSQL's `timestamptz` type, ensuring that temporal records are unambiguous regardless of the geographic location of the server or the timezone in which the operation was performed.

Client-side timestamps — such as the time at which a sample was physically collected — are captured at the point of action and transmitted as part of the event payload. These timestamps are preserved alongside the server-side `createdAt` timestamp, enabling the distinction between the time of the physical operation and the time at which the event was received and persisted by the backend. This is particularly relevant for events created during offline operation, where a delay between physical action and server-side persistence is expected.

The combination of append-only storage, monotonic versioning, timezone-aware timestamps, server-side ordering and client-side temporal metadata ensures that the system can reconstruct an accurate and defensible chronological record of the complete sample lifecycle, from initial collection through custody transfers to analytical evaluation.

6 VALIDATION

6.1 Validation Approach

The development of the sample tracking application followed a rapid prototyping and lean development methodology. Rather than implementing a fully specified system prior to evaluation, successive functional prototypes were developed and incrementally validated through internal testing and targeted end-user feedback.

This approach allowed early identification of usability and workflow issues while maintaining flexibility to refine system functionality during development. Validation activities therefore focused on two complementary aspects: technical verification of implemented features and operational feedback from practitioners familiar with CBRN sampling procedures.

Functional testing was conducted to verify that the core system components operated as intended, while end-user testing sessions were organised to assess the suitability of the application workflow for field use. This iterative validation process supported continuous refinement of the system design and ensured alignment with practical sampling and chain-of-custody procedures.

6.2 Feature Testing Results

Functional testing was conducted to verify that the core operational features of the sample tracking application perform as intended during field use. The testing focused on the main workflow components that support sample documentation, traceability, and digital chain-of-custody management.

The following system capabilities were evaluated during testing:

Sample registration functionality

The application was tested for its ability to register individual samples within an operational mission. This included creating new sample entries and associating them with relevant sampling activities performed in the field.

NFC tag scanning and physical–digital linkage

Testing verified the correct operation of NFC scanning to associate physical sample containers with their corresponding digital records. The functionality ensures that each physical evidence item can be uniquely identified and linked to its digital representation in the system.

Metadata capture during sampling

The application's ability to record structured sample metadata was tested. This included assigning attributes such as sample type, aliquot designation, control sample indication, solvent information, and other operational parameters during the registration process.

Sample aggregation into collections

Testing verified the functionality allowing multiple samples to be grouped into a collection. This included the creation of collections and the association of individual samples with a shared container or packaging unit.

Container sealing and NFC identification

The process of sealing collections and linking them to an NFC-tagged seal identifier was tested. This functionality ensures that packaging procedures can be digitally documented and that sealed containers remain traceable within the system.

Custody transfer recording

The system's capability to document custody transfers between authorised users was evaluated. This included authentication of both the transferring and receiving resources and recording the associated transfer information.

Custody Transfer Point documentation

Testing verified that custody transitions are recorded with the necessary metadata, including user identity, location information, and timestamps. This ensures that each custody transfer is documented in a traceable manner consistent with digital chain-of-custody principles.

6.3 End User Feedback and Workflow Evaluation

To complement the functional testing activities, operational feedback was collected from end users with experience in CBRN sampling procedures. Hands-on testing of the sample tracking application was conducted by experts from VER, who evaluated the application using the current development version. The testers performed representative tasks related to tag scanning, sample registration and workflow navigation in order to assess the operational suitability of the system. The detailed feedback report produced during this testing activity is provided in Annex A.

The testers performed representative tasks related to tag scanning, sample registration and workflow navigation. The objective was to assess whether the implemented functionality aligns with operational sampling practices and whether the user interface supports efficient use in field conditions.

The evaluation confirmed that the core technical concept of the system is sound. In particular, NFC tag detection and sample registration were observed to function correctly when initiated by the user. The testers confirmed that the system is capable of linking physical identifiers with digital sample records, thereby supporting traceability of samples within the chain-of-custody process.

At the same time, several areas for improvement were identified. These observations primarily concerned workflow clarity, user guidance and overall usability of the interface.

The testers noted that some interactions require prior knowledge of the intended workflow, which may reduce usability for new or infrequent users. In several application views, the system does not clearly indicate the current operational state or the next expected action. This may increase cognitive load during operation, particularly in time-critical field environments.

Feedback was also provided regarding the visual layout of certain screens. Some interface elements make use of large, screen-filling buttons, which were likely intended to support operation while wearing protective gloves. While this design consideration is understandable, the testers observed that the large buttons can dominate the interface and reduce the amount of information visible at a glance.

Another observation concerned the NFC scanning workflow. Testers reported that the requirement to manually initiate scanning through a button press was not immediately intuitive. When a tag was presented to the device before initiating the scan function, the system displayed an "empty tag" message, which was considered ambiguous. The testers recommended clearer prompts explaining when scanning is active and what the system expects from the user.

During navigation through the application, testers also identified situations where the system did not automatically progress to the next logical workflow step following user confirmation actions. In some cases, users were left in intermediate states without clear guidance on how to continue the workflow.

Finally, the testers observed several stability issues during testing, including application crashes during basic interactions. These issues limited the extent of testing that could be performed and were identified as an area requiring further technical refinement.

Despite these observations, the testers confirmed that the fundamental concept of the system is appropriate for supporting digital documentation of sampling activities and chain-of-custody processes. The feedback provided through this evaluation has been documented and incorporated into the ongoing development process, informing subsequent iterations of the application's workflow design and user interface improvements.

7 KNOWN LIMITATIONS

The current version of the sample tracking system represents the first operational prototype developed during the initial iteration of the project. While the core architecture and principal workflow components have been implemented and validated, certain technical and functional limitations remain.

These limitations primarily relate to aspects of system maturity, security hardening, extended functionality and operational robustness that are planned to be addressed in subsequent development iterations. Identifying these constraints at this stage provides transparency regarding the current system capabilities and helps guide future development priorities.

The following subsections outline the main known limitations identified during the technical review of the system architecture and implementation.

Unauthenticated analysis endpoints

The document explicitly states that all three analysis endpoints (registration, result submission, attachment upload) are marked as `@Public()` and do not require JWT authentication. This means any party with network access to the API could submit, modify or attach files to analysis records without identity verification. The document itself acknowledges this and identifies API key-based authentication as a planned future enhancement.

In-memory offline storage (no persistence to disk)

The mobile application stores offline-created entities in the Redux in-memory store. If the application is terminated unexpectedly (crash, device restart, battery failure), locally queued samples, sample details and collections that have not yet been synchronised would be lost. There is no mention of persistent local storage (e.g. SQLite or AsyncStorage-backed persistence) to survive application restarts.

Limited offline capability scope

Only samples, sample attribute details and collections can be created offline. Custody transfers, which require QR code scanning and two-party confirmation, appear to depend on connectivity. This means that in disconnected field environments, the custody transfer workflow cannot be executed, potentially delaying the formal chain of custody process.

Single-role assignment per user

Each user is assigned exactly one role. This means a user who needs to act as both a Documenter and a Transporter in the field (which may occur in small-team scenarios) would require two separate accounts or would be unable to perform both functions.

No multi-factor authentication

Authentication relies solely on username and password with JWT tokens. For a system handling forensic evidence with potential judicial implications, the absence of multi-factor authentication (e.g. biometric, hardware token) represents a limitation in identity assurance strength.

Limited sample type categories

The system currently supports only four sample types: wipe, air, solid and liquid. Specialised CBRN sampling categories (e.g. biological swabs, radiological smears, water samples) are not explicitly represented and would need to be mapped into these generic categories, potentially losing specificity.

No event log review or audit interface

While the event store captures a complete immutable history, the document does not describe any user-facing interface for reviewing, querying or exporting the raw event log. Supervisors appear to have access only to projected read models, not to the underlying event stream.

Absence of advanced CEN/TS 18053 features

The document explicitly acknowledges that certain features described in CEN/TS 18053, such as automated KPI monitoring and extended compliance analytics, are not yet implemented.

No conflict resolution for synchronisation edge cases

While UUIDs are generated client-side to prevent duplicates, the document does not describe how the system handles potential logical conflicts (e.g. a sample being added to a collection that was sealed on another device during the offline period).

8 NEXT STEPS FOR 2ND ITERATION

8.1 Planned Improvements

8.1.1 Improvements Identified by End User Feedback

The end-user evaluation described in Section 6.3 identified several usability improvements and feature refinements for the sample tracking application. These observations were analysed and translated into concrete development items to be addressed in the second iteration of the system.

The items described below represent improvements and functional refinements derived directly from tester feedback and are intended to enhance workflow clarity, user interaction and operational reliability during field deployment.

Guided workflow and clearer user prompts

User feedback indicated that some actions within the application require prior knowledge of the intended workflow. Future development will therefore introduce improved workflow guidance within the interface, including clearer prompts, step indicators and more explicit descriptions of the next required action during sampling operations.

Improved NFC scanning interaction

Testing showed that the current requirement to manually initiate the NFC scanning process may not be intuitive for first-time users. Future iterations will therefore improve the scanning interaction by providing clearer instructions and potentially enabling more automated tag detection behaviour when a tag is presented to the device.

Optimised user interface layout

Testers noted that some views contain very large buttons intended to support operation while wearing protective gloves. While this design choice supports field usability, it can reduce the visibility of contextual information. Future improvements will focus on optimising the layout to balance glove-friendly interaction with improved information density and clearer visual hierarchy.

Enhanced tag status feedback

During testing, the application displayed an “empty tag” message when tags were presented before the scan function was activated. This message was considered ambiguous by testers. Future development will therefore improve system feedback by providing clearer status messages indicating whether a tag has been detected, requires registration, or is awaiting user confirmation.

Improved workflow progression

Testers observed that after certain confirmation actions the application did not always automatically proceed to the next logical step in the sampling workflow. Improvements will therefore ensure that user actions consistently trigger the appropriate next stage in the operational process.

Application stability improvements

Several application crashes were observed during testing, limiting the extent of usability evaluation. Improving overall system stability will therefore be prioritised in the next iteration to ensure reliable operation in operational field environments.

The improvements identified through user feedback form a key input for the next development iteration and support the continued refinement of the sample tracking application for operational use in CBRN sampling missions.

8.1.2 Other Improvements

In addition to the improvements identified through end-user feedback, several further enhancements have been identified based on the current system architecture and operational requirements. These improvements focus primarily on extending supervisory capabilities and improving situational awareness for command-level personnel.

The improvements described below will be implemented during the second iteration to expand the capabilities of the Supervisor Dashboard and provide more comprehensive oversight of operational activities.

Extended supervisor dashboard capabilities

The current Supervisor Dashboard provides incident creation and viewing capabilities, which support the initial requirement of establishing incident contexts for field activities. However, it does not yet present the full range of operational data captured by the system. The next development iteration will therefore extend the dashboard to provide more comprehensive operational oversight and real-time situational awareness for supervisory personnel.

Incident timeline visualisation

A key improvement will be the introduction of an incident-level timeline view. This feature will reconstruct and display the chronological sequence of events associated with a given incident, including sample registrations, collection creation and sealing events, custody transfers and analytical results. By presenting these events in a unified chronological interface, supervisors will be able to monitor the progression of field operations and identify potential procedural delays or irregularities.

Comprehensive incident sample overview

The dashboard will be extended to display all samples associated with an incident, including their current status, assigned metadata, NFC identifiers and collection associations. This will allow supervisors to monitor sampling progress, verify that expected sample types have been collected and review the metadata captured during field operations. Where analytical results have been submitted, these will be presented alongside the corresponding sample records to provide an integrated view of the evidentiary and analytical status of each sample.

Custody transfer monitoring

Visibility of custody transfer activities will be introduced as a dedicated dashboard feature. Supervisors will be able to review transfers initiated within an incident, including the identities of the transferring and receiving parties, transfer status, the collections and samples involved, and the outcome of item-level verification. This functionality will support supervisory oversight of the chain of custody and enable early identification of incomplete transfers or verification discrepancies.

Collection composition overview

Collection-level views will be added to present the composition of each sealed collection. These views will display the samples contained within a collection, the sealing timestamp and the identity of the user responsible for sealing the container. This will allow supervisors to verify that collections have been assembled and sealed in accordance with established operational procedures.

9 CONCLUSION

This deliverable has presented the first iteration of the EMBRACE sampling and sample tracking capability developed under WP3 Task 3.1. The work establishes a structured technical and operational baseline for the digital documentation, traceability and governance of biological and CBRN-related sampling activities. It combines an established wipe-sampling reference methodology with a digital CBRN Sample Tracking System designed to preserve evidential continuity from field collection through custody transfer and subsequent analysis.

A principal outcome of this first iteration is the successful definition and implementation of a multi-component system architecture consisting of a mobile field application, a Supervisor Dashboard and a backend API. Together, these components provide the core functionality required to register samples within an incident context, capture structured metadata, bind physical items to digital records through NFC, group samples into sealed collections, record custody transitions and link analytical activity to the relevant specimen. In this respect, the deliverable demonstrates that EMBRACE has moved beyond conceptual design and has produced an operational prototype capable of supporting traceable sampling workflows in CBRN environments.

The deliverable also shows that the system has been designed in close alignment with established forensic and CBRN doctrine. Rather than introducing a new sampling methodology, the EMBRACE solution reinforces recognised operational practice by digitising the documentation and custody controls required for defensible sample handling. In particular, the implementation of role-based responsibilities, structured Digital Custody Metadata and explicit Custody Transfer Points reflects the principles of CEN/TS 18053 and supports accountability, non-repudiation and interoperability across the sampling lifecycle. This is a significant contribution to the project's wider objective of harmonising evidence governance for biological toxin incidents at European level.

From a technical perspective, the first iteration establishes several strong foundations for future development. These include the event-based architecture supporting immutable traceability, the OpenAPI-based interface definition for system consistency and integration, the role-based access model, and the generic analysis endpoint structure enabling interoperability with external analytical devices. The design choices documented in this deliverable position the system as a flexible and extensible digital backbone capable of integrating both operational sampling actions and downstream analytical reporting within a unified evidentiary framework.

At the same time, the validation activities confirm that this remains a first operational prototype rather than a fully mature deployment-ready platform. End-user evaluation verified that the core concept is sound and that essential functions such as NFC-based sample registration operate correctly. However, the testing also identified important improvements relating to workflow clarity, user prompting, interface usability and application stability. These findings are valuable because they provide practical evidence from intended users and ensure that the second iteration will respond not only to architectural requirements but also to operational realities in the field.

The document is appropriately transparent regarding current limitations. Several constraints remain, including unauthenticated analysis endpoints, the absence of persistent offline storage, limited offline support for custody transfer workflows, single-role assignment per user, lack of multi-factor authentication, restricted sample type granularity, absence of a user-facing audit log interface,

incomplete implementation of advanced CEN/TS 18053 features, and unresolved synchronisation conflict scenarios. These limitations do not invalidate the prototype; rather, they define the maturity boundary of the first iteration and provide a clear agenda for technical hardening and functional expansion in the next development phase.

The planned next steps are therefore both credible and necessary. The second iteration will improve guided workflows, NFC interaction, system feedback, interface layout and application stability, while also extending the Supervisor Dashboard with timeline reconstruction, broader incident-level sample visibility, custody transfer monitoring and enhanced collection oversight. These developments will strengthen both field usability and command-level situational awareness, thereby moving the EMBRACE solution closer to an operationally robust and reviewer-defensible capability for biological incident response.

D3.1 sets out the first iteration of the EMBRACE sampling and digital chain-of-custody capability. It defines the main methodological, architectural and procedural elements required to support traceable biological sample management in complex CBRN contexts. The results presented here provide the basis for the second iteration, which will address identified needs relating to usability, resilience, security and supervisory functionality. In this respect, the deliverable constitutes an initial step towards the EMBRACE objective of supporting harmonised, interoperable and evidentially robust management of biological toxin incidents across Europe.

10 REFERENCES

CEN. 2024. CEN/TS 18053-1:2024 – Digital Chain of Custody for CBRNE Evidence – Part 1: Overview and Concepts. European Committee for Standardization, Brussels.

CEN. 2024. CEN/TS 18053-2:2024 – Digital Chain of Custody for CBRNE Evidence – Part 2: Data Management and Audit. European Committee for Standardization, Brussels.

EuroBioTox Consortium. 2023. D6.3 Guidelines for First Responders: Sampling in a Biological Toxin Incident. Horizon 2020 Project EuroBioTox.

NATO (2014). AEP-66: NATO Handbook for Sampling and Identification of Biological, Chemical and Radiological Agents (SIBCRA). Edition A, Version 1. NATO Standardization Office (NSO).

TOXI-Triage Consortium. 2017. D2.1 Manual on CBRN Sampling Protocols. Horizon 2020 Project TOXI-Triage.

Polish Academy of Sciences (or appropriate institution). 2023. Organisation of the Biological Sampling Process. Project internal guidance document.

VERIFIN. 2026. User Feedback Report on EMBRACE Sampling and Chain-of-Custody Application. EMBRACE Project, Annex A.

ANNEXES

Annex A. VER App User Feedback Report

PRO – Tag-and-Trace App

Test Report and Workflow Feedback

Testers: VERIFIN Matti Kuula, Martin Hermansson.
Date: 03 Feb 2026
Application: Tag-and-trace app (current development version)
Context: Functional and usability testing for sample handling and chain-of-custody workflows

---

Summary

Testing of the current tag-and-trace app shows that the essential building blocks for NFC-based tag detection and sample registration are already in place. Tag reading and registration function as intended when initiated, indicating that the core technical approach is sound.

The main opportunities for improvement relate to workflow clarity, user guidance, and overall intuitiveness, particularly for first-time or field users. Some views rely on very large, screen-filling buttons, likely designed to support operation with gloved hands. While this design choice is understandable, in practice the buttons can visually dominate the screen, effectively dividing it into large regions rather than clearly defined actions. This can make it harder to perceive the available options and their relative importance at a glance.

Several usability issues, such as ambiguous handling of “empty tags”, unclear progression and screen layouts appear to be consequences of an implicit structure rather than fundamental design flaws.

This report therefore combines targeted feedback on the current implementation with a **proposed conceptual workflow** intended to support future iterations. The suggested improvements focus on making system state and next actions more explicit, simplifying user interactions, and incrementally refining the UI, without requiring a complete redesign of the application.

1. Scope and purpose

This test had two objectives:

1. To provide feedback on the current implementation based on hands-on testing
2. To propose constructive workflow improvements to support future development

The intention is to identify practical improvements and to clarify a workflow model that may guide further iterations.

2. Feedback on the current version

2.1 Usability and intuitiveness

- The application is currently **not fully intuitive**, particularly for new or infrequent users
- Several actions require prior knowledge of the intended workflow
- In multiple views, the app does not clearly indicate:
    - The current state
    - The expected next action

Impact: Increases cognitive load and slows down operation, especially in field or time-critical use.

2.2 Screen layout and interaction density

- Some views contain too many options per page
- Buttons are relatively large, reducing information density and overview

Impact:

- The interface feels heavier than necessary for task-oriented workflows

Suggested incremental improvements:

- Reduce the number of simultaneous options
- Use smaller, more compact buttons where appropriate
- Apply progressive disclosure (show options only when relevant)

2.3 Tag scanning and NFC behaviour

- Tag detection and registration **works** when explicitly initiated via the "Scan tag" button and confirmed by the user
- Tag information is correctly registered under these conditions

Observed issues:

- Manual scan initiation is unintuitive (the testers initially attempted to scan the tag when the scan tag prompt/button was displayed)
- When doing this, the app displays an "empty tag" indicator, which is ambiguous and not self-explanatory

In particular, it is unclear whether an empty tag is:

- Undetected
- Detected but uninitialized
- Awaiting user action

Solution: Add further information to the **scan tag**-button (e.g. press button to proceed)

2.4 Navigation and workflow continuity

- After certain actions (e.g. tag confirmation), the app does not consistently advance to the next logical step
- Users may end up in dead-end states
- Returning to the main workflow is not always obvious

Impact:
Disrupts task flow and increases the risk of user error.

2.5 Stability

- The application was observed to be **unstable**
- Crashes occurred during basic interaction, limiting further testing

Impact:
Stability issues currently overshadow otherwise functional components.

3. Suggested workflow improvements (forward-looking)

In the tester's view, the implemented **workflow hierarchy, while functional,** may hinder applicability in the hot zone. A more straight-forward, guided, process might make the app more intuitive.

The following workflow model is intended as **conceptual suggestions** for future development.

3.1 Core design principle

The workflow **should be initiated automatically by bringing the scanning device next to a tag**. This reduces unnecessary manual interaction and aligns better with field use.

3.2 Proposed workflow (conceptual)

Entry point

Trigger: Device brought into NFC range of a tag

1. **Immediate tag read**
    - Display TAG ID and basic tag information
2. **Tag status indication**
    - "Tag detected – ready to be registered"
    - or "Tag already registered"
3. **Contextual prompt**
    - New tag → *Register new sample*
    - Registered tag → *Proceed to packaging*

3.3 Sample registration flow

- Sample type (e.g. wipe / swab)
- Solvent information (with / without solvent)
- Additional sample details (tiered)
- Optional free-text / metadata
- Finish entry (sample registered and linked to tag)

3.4 Packaging flow

- Insert package information
- Add additional samples (loop via next tag scan)
- Seal package (final confirmation)

3.5 UI/UX alignment

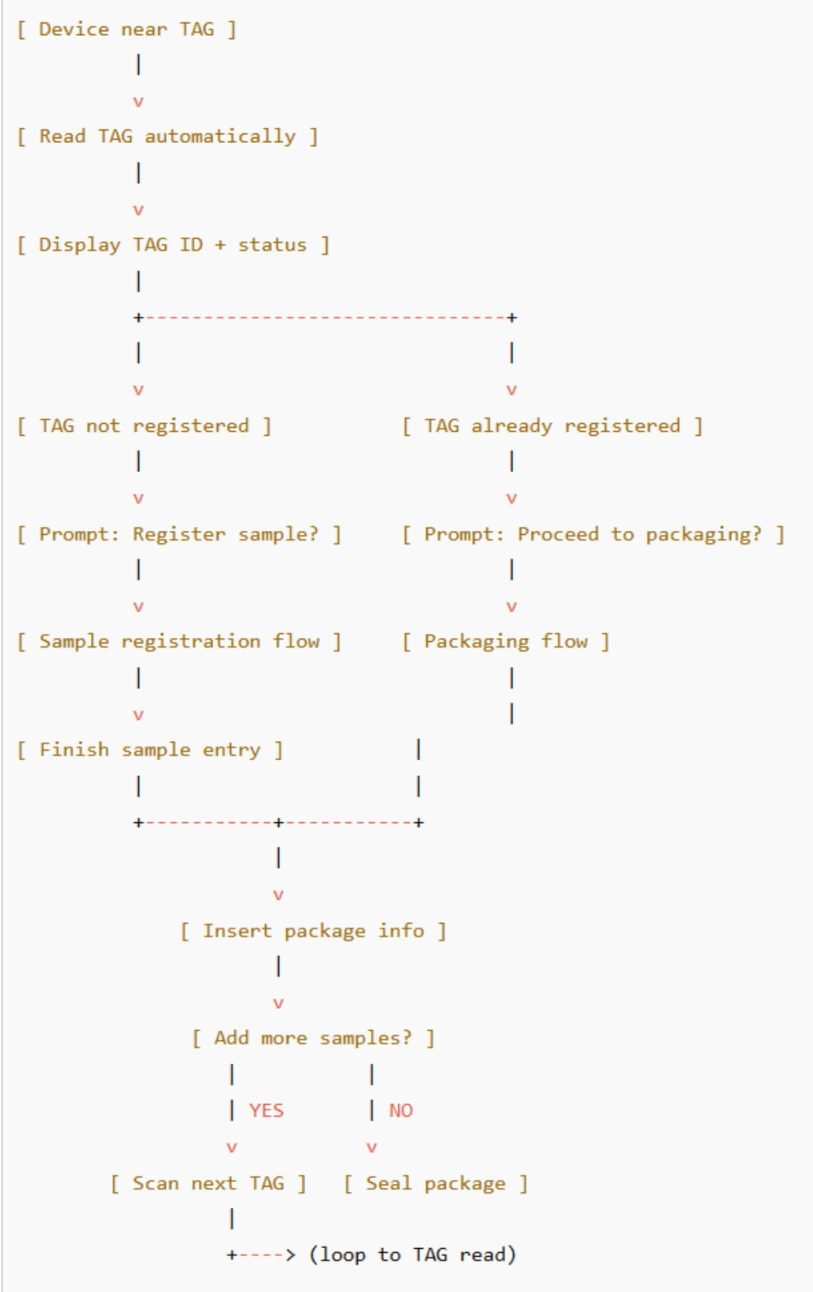
To support the above workflow:

- Limit options per screen
- Use smaller, contextual buttons
- Keep key information always visible:
    - TAG ID
    - Current workflow stage
- Prefer prompts and guided actions over static option lists

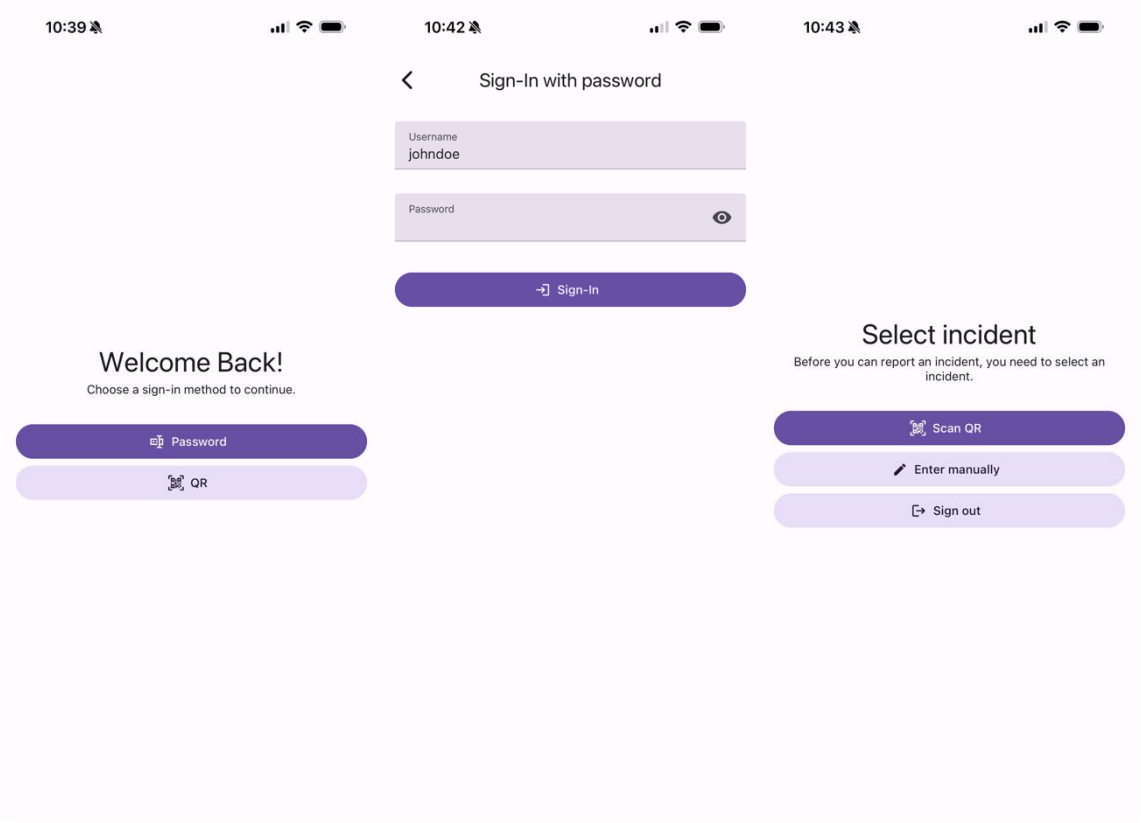
4. Concluding remarks

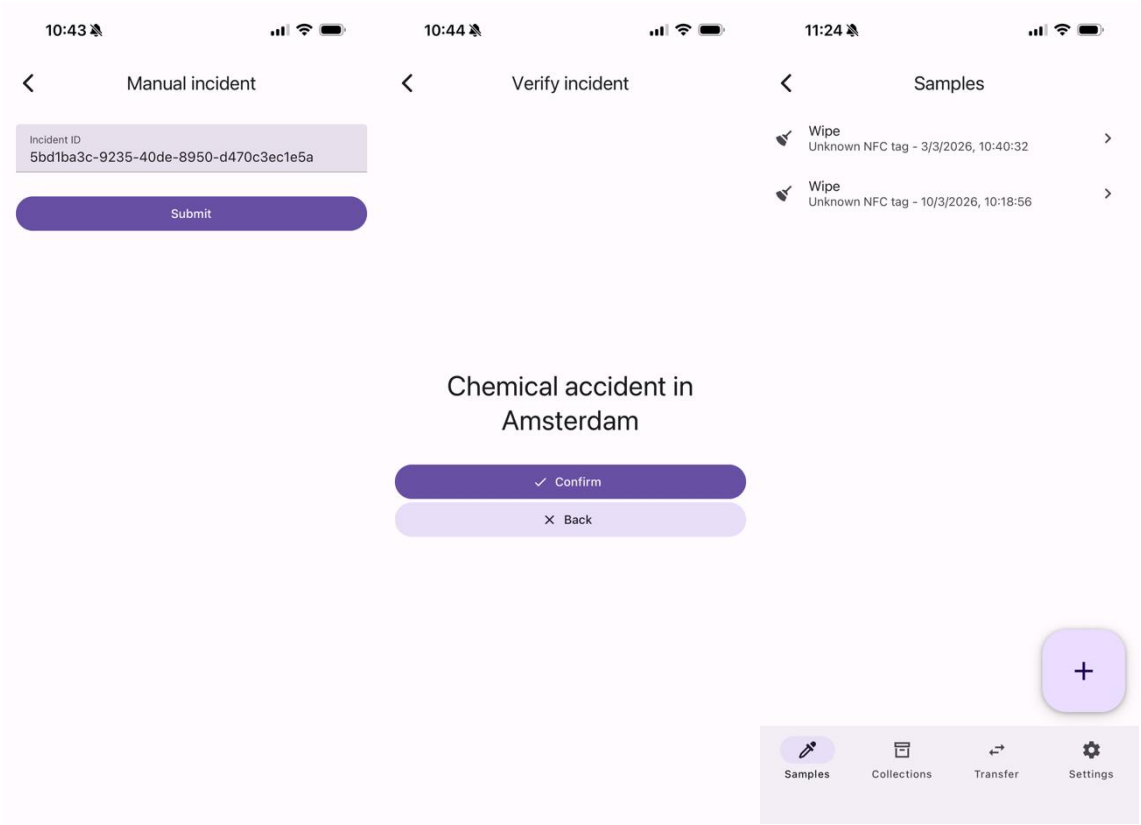
Core functionality for tag detection and registration is already present. However, improvements in workflow clarity, UI design, and stability would improve effective field deployment. Many of the observed issues are likely resolvable through clearer workflow definition and incremental UI/UX refinements.

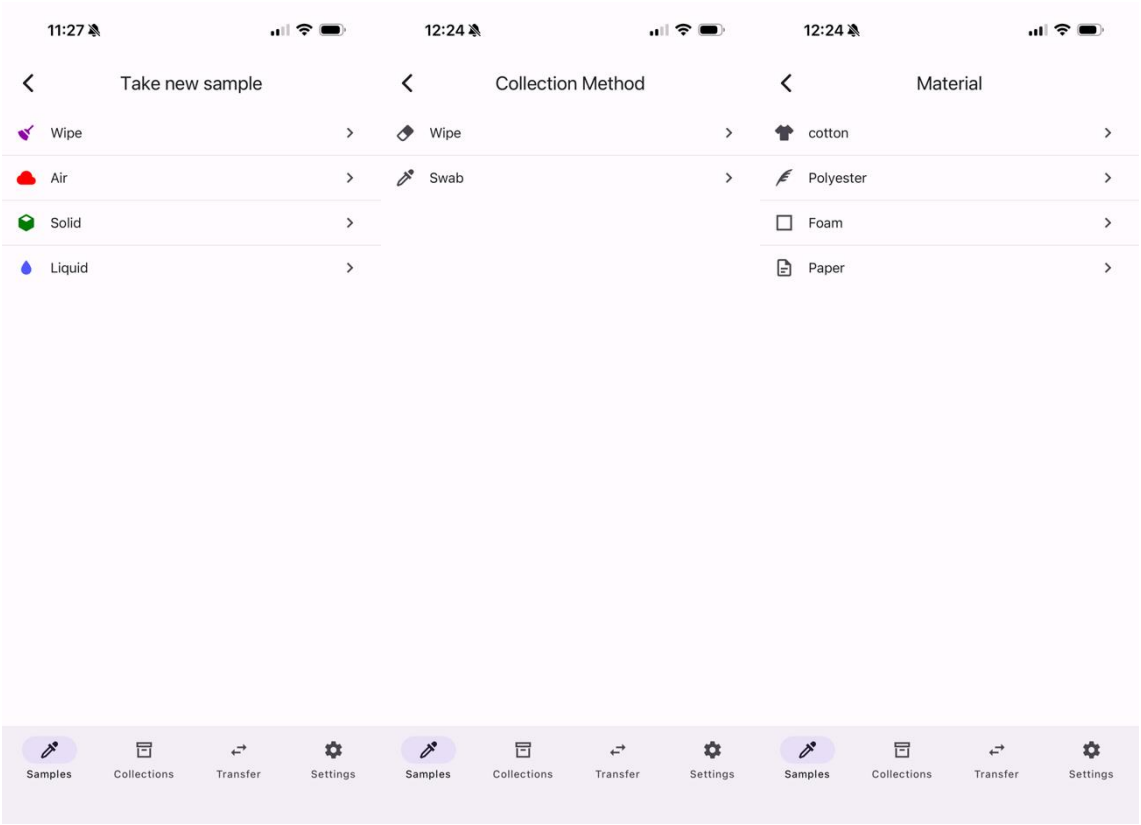
Appendix A – Conceptual Workflow Flowchart

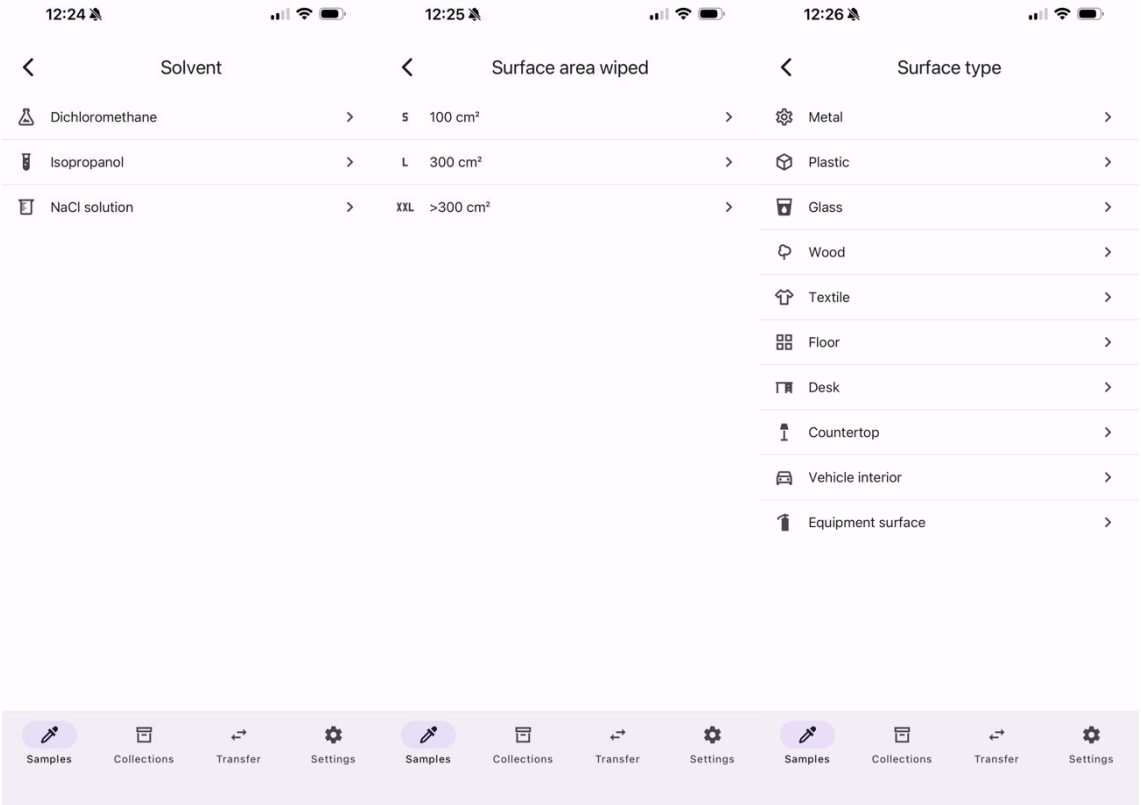


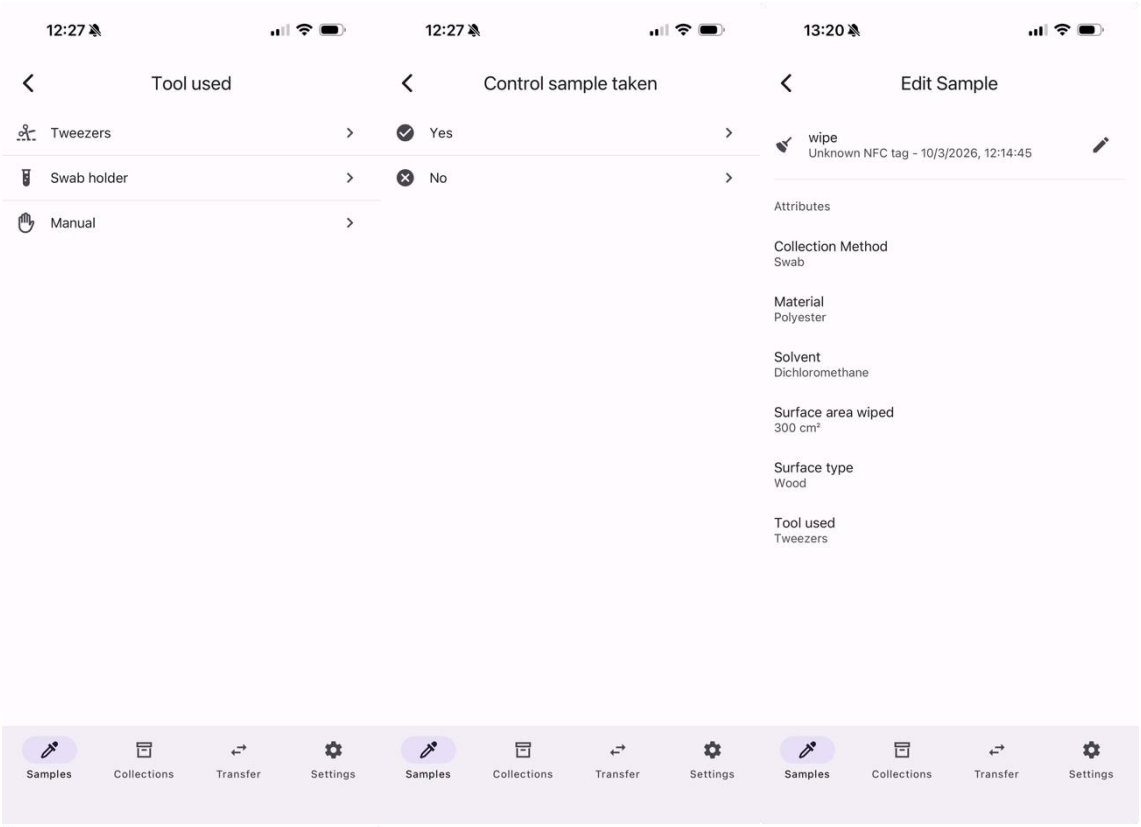
Annex B. User Interface Screenshots

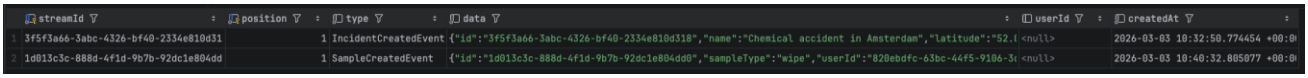





11:27
Take new sample
Wipe
Air
Solid
Liquid
Samples
Collections
Transfer
Settings
12:24
Collection Method
Wipe
Swab
Samples
Collections
Transfer
Settings
12:24
Material
cotton
Polyester
Foam
Paper
Samples
Collections
Transfer
Settings


12:24
Solvent
Dichloromethane
Isopropanol
NaCl solution
Samples
Collections
Transfer
Settings
12:25
Surface area wiped
S 100 cm²
L 300 cm²
XXL >300 cm²
Samples
Collections
Transfer
Settings
12:26
Surface type
Metal
Plastic
Glass
Wood
Textile
Floor
Desk
Countertop
Vehicle interior
Equipment surface
Samples
Collections
Transfer
Settings



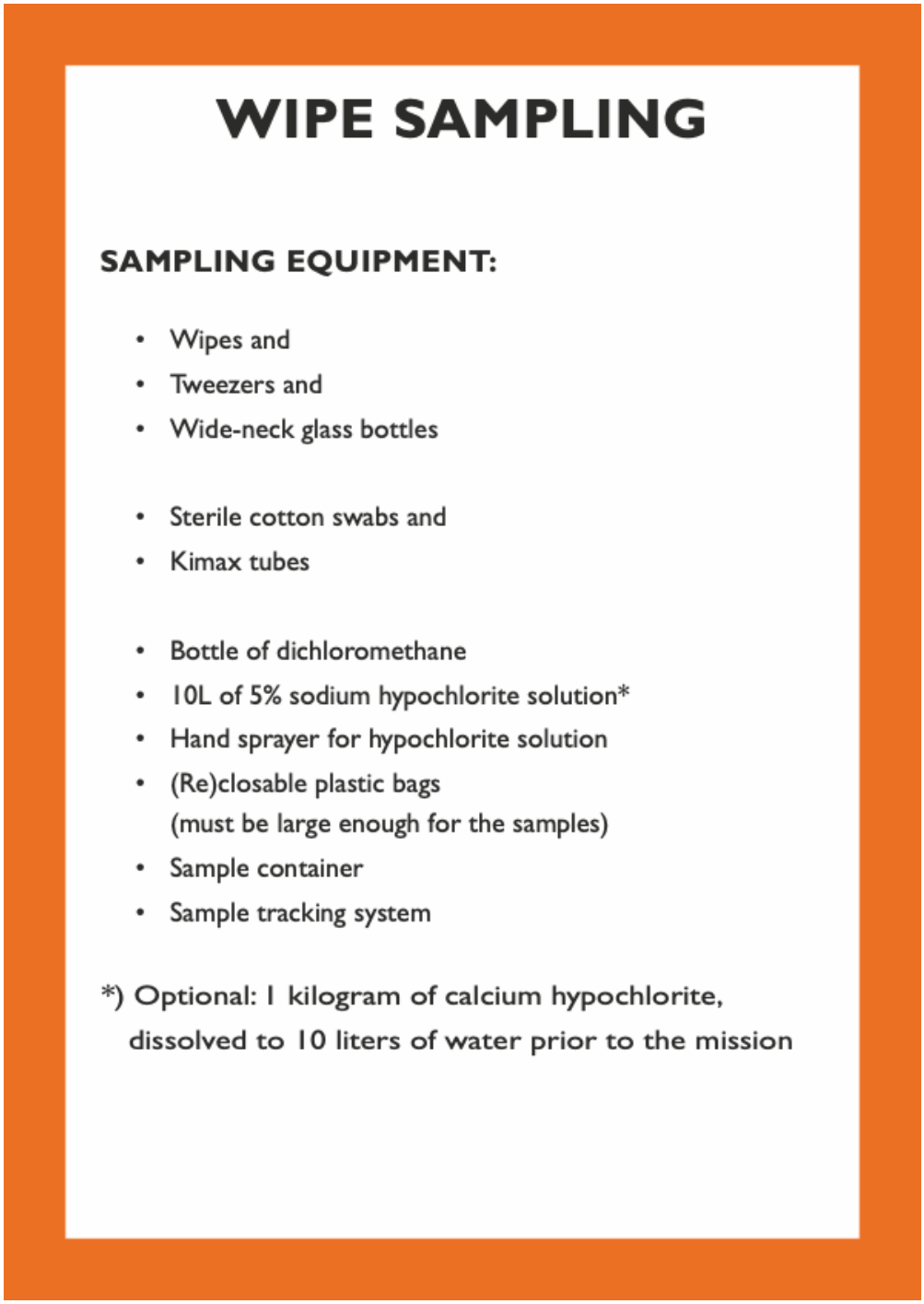


streamId	position	type	data	userId	createdAt
3f5f3a66-3abc-4326-bf40-2334e810d31	1	IncidentCreatedEvent	{"id":"3f5f3a66-3abc-4326-bf40-2334e810d318","name":"Chemical accident in Amsterdam","latitude":"52.(	<null>	2026-03-03 10:32:50.774454 +00:0
1d013c3c-888d-4f1d-9b7b-92dc1e804dd	1	SampleCreatedEvent	{"id":"1d013c3c-888d-4f1d-9b7b-92dc1e804dd0","sampleType":"wipe","userId":"820ebdfc-63bc-44f5-9186-3(	<null>	2026-03-03 10:40:32.805077 +00:0

	streamId	position	type	data	userId	createdAt
1	3f5f3a66-3abc-4326-bf40-2334e810d31	1	IncidentCreatedEvent	{"id":"3f5f3a66-3abc-4326-bf40-2334e810d318","name":"Chemical accident in Amsterdam","latitude":"52.(	<null>	2026-03-03 10:32:50.774454 +00:0
2	1d013c3c-888d-4f1d-9b7b-92dc1e804dd	1	SampleCreatedEvent	{"id":"1d013c3c-888d-4f1d-9b7b-92dc1e804dd0","sampleType":"wipe","userId":"820ebdfc-63bc-44f5-9106-3(	<null>	2026-03-03 10:40:32.805077 +00:0
3	1d013c3c-888d-4f1d-9b7b-92dc1e804dd	2	SampleAttributeAssignedEvent	{"sampleId":"1d013c3c-888d-4f1d-9b7b-92dc1e804dd0","attributeId":"585d1b9b-2a96-40d8-a299-7aaaf4aa7d(	<null>	2026-03-10 11:37:53.645265 +00:0
4	1d013c3c-888d-4f1d-9b7b-92dc1e804dd	3	SampleAttributeAssignedEvent	{"sampleId":"1d013c3c-888d-4f1d-9b7b-92dc1e804dd0","attributeId":"cf1f4825-0658-4ea6-a14e-5d409dc34a(	<null>	2026-03-10 11:37:57.229711 +00:0
5	7b8b4714-9a24-47d2-b584-bd9768a8d48	1	SampleCreatedEvent	{"id":"7b8b4714-9a24-47d2-b584-bd9768a8d487","sampleType":"wipe","userId":"820ebdfc-63bc-44f5-9106-3(	<null>	2026-03-10 12:14:45.105214 +00:0
6	7b8b4714-9a24-47d2-b584-bd9768a8d48	2	SampleAttributeAssignedEvent	{"sampleId":"7b8b4714-9a24-47d2-b584-bd9768a8d487","attributeId":"7f707583-6b9c-4dd1-9b48-192c5bc4de(	<null>	2026-03-10 12:14:46.906850 +00:0
7	7b8b4714-9a24-47d2-b584-bd9768a8d48	3	SampleAttributeAssignedEvent	{"sampleId":"7b8b4714-9a24-47d2-b584-bd9768a8d487","attributeId":"1fd14098-e95a-45b5-98f2-5a32a841d0(	<null>	2026-03-10 12:14:48.358581 +00:0
8	7b8b4714-9a24-47d2-b584-bd9768a8d48	4	SampleAttributeAssignedEvent	{"sampleId":"7b8b4714-9a24-47d2-b584-bd9768a8d487","attributeId":"4d2917a5-cdcb-43ee-8a3a-5a65d719ab(	<null>	2026-03-10 12:14:50.038845 +00:0
9	7b8b4714-9a24-47d2-b584-bd9768a8d48	5	SampleAttributeAssignedEvent	{"sampleId":"7b8b4714-9a24-47d2-b584-bd9768a8d487","attributeId":"3248d782-85c5-4e05-bf30-57e945a051(	<null>	2026-03-10 12:14:51.315171 +00:0
10	7b8b4714-9a24-47d2-b584-bd9768a8d48	6	SampleAttributeAssignedEvent	{"sampleId":"7b8b4714-9a24-47d2-b584-bd9768a8d487","attributeId":"7a8204bb-6c35-42d8-aa1e-865dede508(	<null>	2026-03-10 12:14:58.702731 +00:0
11	7b8b4714-9a24-47d2-b584-bd9768a8d48	7	SampleAttributeAssignedEvent	{"sampleId":"7b8b4714-9a24-47d2-b584-bd9768a8d487","attributeId":"95afb7f6-b4e0-4a24-b53c-5f9de09b2d(	<null>	2026-03-10 12:15:00.518514 +00:0

Annex C. Wipe Sampling Procedure Documentation

Source: VER



Annex D. Enlarged Sample Collection Activity Diagram