UN-Habitat Tenure Rights Registration and Claims Management System

Functional Specification Document (FSD)

Project Name	**UN-Habitat Tenure Rights Registration and Claims Management System**
Client	UN-Habitat Aleppo
Prepared By	VRoad LLC
Approved By	UN-Habitat Program Manager
Prepared On	2025-12-20

Version: 5.0

Date: 2025-12-20

1. Executive Summary

This FSD defines WHAT the UN-Habitat Tenure Rights Registration & Housing, Land and Property (HLP) solution must do across two components: (1) an offline-first Android mobile app built with Flutter used by field data collectors in areas without reliable connectivity, and (2) a Windows desktop application built with Python and PyQt5 for back-office case management, quality control, and claims processing at the municipality offices.

The design fulfills three core program recommendations: provide robust software capable of data extraction and analysis; assign distinct identifiers for Buildings, Units, and Claims (Building ID, Unit ID, Claim ID); and deliver comprehensive search and reporting features.

Key Features

- Offline-first mobile data collection
- Unique 17-digit Record ID system and other IDs such as Building ID, Claim ID
- Arabic-first UI and reporting
- Comprehensive audit trail
- QGIS/PostGIS spatial analysis
- Controlled vocabulary management
- Survey reference code generation for interviewees
- Field team building assignment and tablet transfer

2. Goals & Scope

Goals

1. Data Collection Excellence

- Capture high-quality field data offline
- Support Arabic data entry and validation
- Manage photos and documents efficiently

2. Data Integrity

- Generate unique Record IDs
- Detect and resolve duplicates
- Maintain complete audit trail

3. Claim Management

- Register new claims (both field and office)
- Capturing, validating, and classifying ownership/residency evidence
- Update existing claim records
- Correct errors in previously recorded data
- Track modification history for all changes
- Identify conflicting rights claims to the same property

4. Analysis & Reporting

- Provide comprehensive search and filtering
- Generate Arabic PDF reports

- Export to standard formats (CSV/Excel/PDF/GeoJSON)
- RESTful API access with value codes alongside display names

In Scope

- Android tablet application (Flutter) for field data collection with full synchronization capability when connected to same network as desktop
- Windows desktop application (Python, PyQt5):
  - Direct claim registration at municipality offices
  - Import and validation of field-collected data from tablets
  - Management of persons of concern with classification and flagging
  - Case classification and prioritization features
  - Assign buildings to field teams and transfer to tablets
- RDBMS: PostgreSQL 16 + PostGIS database
- Building polygons/points, spatial unit linkage, map-based validation, and integration with existing Aleppo digital spatial archive
- GIS system: QGIS stable long term version
- Controlled vocabulary management
- Role-based access control
- Arabic UI and reports
- Network-based synchronization between tablets and desktop when connected
- Comprehensive dashboards and data extraction capabilities
- RESTful API for system integrations
- Data export APIs for external analytics (QGIS, Power BI, STDM mapping)

Out of Scope

- Automated cloud synchronization from tablets
  - Rationale: System operates without requiring internet connectivity. Data transfer is accomplished through two supported methods:
    - Manual transfer of signed .uhc container files with full auditability
    - Local-network synchronization when tablets and desktop are on the same LAN/Wi-Fi network
  - Note: Local-network sync uses a secure endpoint exposed by the desktop application and does not require external internet connectivity
- Third-party integrations beyond standard exports and documented APIs
  - Rationale: Focus on CSV/Excel/PDF/GeoJSON exports and RESTful API endpoints for maximum flexibility, with value codes alongside display names

3. Stakeholders & Roles

Primary Stakeholders

#	Role/Organization	Responsibilities
1	UN-Habitat Team	Overall program oversight, requirements approval
2	UN-Habitat Technical Consultant	Technical requirements validation, system acceptance, UAT coordination
3	UN-Habitat Field Operations	Field data collection coordination, tablet management, training

#	Role/Organization	Responsibilities
4	Data Managers	Data import, validation, quality assurance, report generation
5	Office Clerks	Front desk claim registration, document scanning, receipt generation
6	Field Supervisors	Monitor field teams, assign tasks, review data quality, coordinate re-visits
7	Field Data Collectors	Collect property and beneficiary data in the field using tablets
8	Local Authorities (Aleppo Municipality)	Provide administrative codes, validate property data
9	Beneficiaries/Claimants	Provide documentation, make property claims

System Access Levels

Role	Mobile Access	Desktop Access	Permissions
Field Data Collector	Full	None	Create/edit cases offline, capture photos/documents, export containers
Field Supervisor	None	Read-only	Monitor field operations, export containers for review, generate field performance reports
Office Clerk	None	Full	Register new claims, update existing claims, scan documents, print receipts
Data Manager	None	Full	Import containers, review/resolve conflicts, finalize cases, generate reports
Analyst	None	Read-only	Run filters/analytics, export CSV/Excel/GeoJSON, print signed PDFs
Administrator	None	Full	Manage users/roles, security policies, configure forms/fields, manage attachments, control vocabularies

4. System Overview

Architecture Components

The system consists of two main applications:

- Mobile Application: Built with Flutter for Android tablets
- Desktop Application: Built with Python and PyQt5 for Windows

Data Flow Process

1. Field data collection
    - Planning: to add/populate Building Reference IDs
    - Field Collection: Data captured offline on tablets
    - Survey reference code generated and handed to interviewee
2. Populating the collected data to the database

    - Export: Generate signed .uhc container file
    - Transfer: Controlled physical transfer to desktop
    - Import: Validate and stage data including completeness checks and deduplication
    - Commit: Generate Record IDs and save to database
3. Reporting: Generate reports and analytics

5. Assumptions & Constraints

5.1 Network & Connectivity

Assumption: The system supports offline-first operations and does not rely on internet connectivity. Local-network synchronization is strictly limited to LAN/Wi-Fi environments and is not a cloud service.

Implementation:

- Tablets operate without network connectivity during field operations
- Data leaves devices via two methods:
  - Exported .uhc files with controlled transfer and chain-of-custody
  - Local-network synchronization when connected to the same LAN/Wi-Fi as the desktop

Explanation: Controlled transfer means recording chain-of-custody (package_id, checksum, operator, time), and ingesting it on a controlled workstation that verifies signature/hash before staging.

5.2 Technology Stack

Constraint: Central DB is PostgreSQL with PostGIS extension; desktop client is Python with PyQt5; mobile client is Flutter. Database must be compatible with older PostgreSQL versions to ensure deployment flexibility; GIS System: QGIS stable LTR version.

Explanation: PostGIS is required for spatial filters (within polygon/buffer), proximity-based duplicate detection for properties, overlay analyses, and direct GeoJSON export—without depending on third-party GIS services.

5.3 Data Governance

Constraint: All vocabularies are centrally governed and versioned; imports enforce compatibility.

Explanation: Code lists (e.g., ownership_type, document_type) use semantic versions MAJOR.MINOR.PATCH. Desktop import checks .uhc manifest vocab_versions; MAJOR mismatches are quarantined, MINOR additions are flagged for review.

5.4 Additional Assumptions

- Device Storage: Tablets have minimum 32GB storage for offline operations
- Office Infrastructure: Municipality offices maintain local network infrastructure
- User Training: All operators receive appropriate training before system access
- Data Sovereignty: All data remains within Syrian jurisdiction

5.5 Additional Constraints

- Platform Requirements:
  - Mobile: Android 8.0+ tablets only
  - Desktop: Windows 10/11 with Python 3.10+ and PyQt5
- Language: Arabic as primary language with full RTL support
- Compliance: Must align with Syrian regulations and UN-Habitat STDM standards

- Performance: System must handle 1M+ records with acceptable response times

6. Data Model & Entities

6.1 Core Entities

6.1.1 Building Management (إدارة البناء)

Purpose: Register and manage building information in the system

Spatial Entity: Buildings are spatial entities represented as either polygon or point geometries in the GIS layer.

Important Note: Each building has both a business identifier (17-digit UN-Habitat code) and an internal technical UUID. The UUID is for system operations and SHALL NOT replace the UN-Habitat coding scheme.

Fields:

- building_uuid: Internal technical identifier (system-generated)
- building_id (رقم البناء): Unique 17-digit identifier (UN-Habitat business code)
- governorate_code: (رمز المحافظة) 2 digits
- district_code: (رمز المنطقة الإدارية) 2 digits
- sub-district code: (رمز الناحية الإدارية) 2 digits
- community_code (رمز التجمع العمراني): 3 digits
- neighborhood_code: (رمز الحي) 3 digits
- building_number: (رقم البناء) 5 digits
- building_geometry: PostGIS geometry (polygon or point)
- Building location: (governorate_name, district_name, subdistrict_name, community_name, neighborhood_name)
- building_type (نوع البناء): Dropdown (residential, commercial, mixed-use houses/shops)
- number_of_property_units (عدد الوحدات العقارية): Integer - total property units in building
- number_of_apartments (عدد المقاسم في البناء): Integer - residential units only
- number_of_shops (عدد المحلات): Integer - commercial units
- building_status (حالة البناء): Dropdown
- geo_location: GPS coordinates (required for point geometry)

6.1.2 Property Unit Management (إدارة الوحدات العقارية)

Purpose: Register property units (apartments, shops, etc.) within buildings. Each building is related to a number of property units.

Relationship: Each property unit belongs to exactly one building (1:N relationship)

Fields:

- unit_uuid: Internal technical identifier (system-generated)
- unit_id (رقم الوحدة): Unique within building
- building_id: Foreign key to building (required)
- unit_type (نوع الوحدة): Dropdown (apartment, shop, office, warehouse, etc.)
- floor_number (رقم الطابق): Integer
- apartment_number (رقم المقسم): String
- property_description (وصف العقار): Text field for detailed property description
- apartment_status (حالة المقسم): Dropdown

6.1.3 Person Management (إدارة الأشخاص)

Purpose: Base entity for all persons in the system using Household multi-layer profiling (owners, occupants, guardians, etc.)

Fields:

- person_id: UUID
- first_name (الاسم الأول): String (Arabic)
- father_name (اسم الأب): String (Arabic)
- mother_name (اسم الأم): String (Arabic)
- last_name/family_name (الكنية): String (Arabic)
- nationality: Dropdown (controlled vocabulary)
- sex/gender: Dropdown (controlled vocabulary)
- is_contact_person (هل هو الشخص المسؤول عن التواصل؟): Boolean
- year_of_birth (سنة الميلاد): Integer - year only, not full date of birth
- national_id (الرقم الوطني): String (validated format - 11 digits)
- id_type: Type of identification document
- id_document_image: Upload identification document image (scanned if available)
- primary_phone (رقم الهاتف الأساشي): String
- secondary_phone (رقم الهاتف الثانوي): String (optional)
- email: String (optional)
- relation_type: Initial relation type to property (owner, occupant, tenant, guest, heirs, other)

6.1.4 Person–Unit Relations (علاقات الشخص-الوحدة)

Purpose: Register and manage all relations between Persons and Property Units (ownership, occupancy, tenancy, guests, heirs, others) in a single structure.

Fields:

- person_unit_relation_id: UUID – unique identifier
- person_id: Foreign key to Person (required)
- property_unit_id: Foreign key to Property Unit (required)
- relation_type (نوع العلاقة): Code (Values: owner, occupant, tenant, guest, heirs, other)
- relation_type_other_description: Short text, required when relation_type = OTHER
- relation_start_date (تاريخ بداية العلاقة): Date
- tenure_contract_type (نوع وثيقة الإشغال/العقد): Code from vocabulary
- ownership_share (حصة الملكية): Integer (0–2400)
- relation_notes (ملاحظات العلاقة): Free-text field
- relation_evidence_ids: Array of Evidence IDs

6.1.5 Evidence (الدليل / البيّنة)

Purpose: Represent each piece of proof that supports a Person–Property Unit Relation.

Main fields:

- evidence_id: UUID – primary key
- reference_number: String
- reference_date: Date

- person_unit_relation_id: UUID – FK to Person–Property Unit Relation (required)
- document_id: UUID – FK to Documents (nullable for non-document evidence)
- evidence_type: Code (document, witness_statement, verbal_claim, photo_only)
- evidence_description: Short human-readable summary
- verification_status: Code (PENDING, VERIFIED, REJECTED)

6.1.6 Occupancy Details (تفاصيل الإشغال)

Purpose: Register households occupying property units with composition details

Fields:

- household_id: UUID
- property_unit_id: Foreign key to Property Unit (required)
- main_occupant_name: String (name of head of household)
- occupancy_size (عدد أفراد الأسرة): Integer
- occupancy_composition (تكوين الأسرة): Structured breakdown with separate categories:
  - By gender: male_count, female_count
  - By age: minors_count (under 18), adults_count, elderly_count (over 65)
  - By disability: with_disability_count, no_disability_count
- occupancy_type (نوع الإشغال): Dropdown
- occupancy_nature (طبيعة الإشغال): Dropdown
- occupancy_start_date: Date
- monthly_rent: Decimal (if rented)
- notes: Optional notes about household

6.1.7 Claim / Case (المطالبة / الحالة)

Concept:

- The base entity is Person–Property Unit Relation
- A Claim is a Person–Property Unit Relation that is still in process
- It has extra workflow/case fields (status, type, source, assignment)

Fields:

- case_number: Human-readable claim/case number
- source: FIELD_COLLECTION, OFFICE_SUBMISSION, SYSTEM_IMPORT
- case_status: Draft, Submitted, Initial Screening, Under Review, Awaiting Documents, Conflict Detected
- Initial status logic: 'Draft' when evidence present, 'Awaiting Documents' when evidence missing

6.1.8 Referral & Lifecycle Management

Purpose: Manage the complete lifecycle of cases/claims.

Lifecycle Stages: Draft → Submitted → Initial Screening → Under Review → Awaiting Documents → Conflict Detected

Referral Actions: Refer Back to Field Team, Refer to Municipality Clerk, Refer to Supervisor

Each referral stores: from_role, to_role, referral_reason, referral_notes, referral_date

6.2 Supporting Entities

6.2.1 Documents (الوثائق)

Fields:

- document_id: UUID
- document_type (نوع الوثيقة): Multi-select dropdown
- issue_date (تاريخ الإصدار): Date
- document_number (رقم الوثيقة): String
- verified (موثق): Boolean
- attachment_hash: SHA-256 reference

6.2.2 Survey (استطلاع ميداني)

Purpose: Track field survey visits and generate reference codes for interviewees

Fields:

- survey_id: UUID
- building_id: Foreign key to Building
- unit_id: Foreign key to Property Unit
- field_collector_id: Foreign key to User
- survey_date: Date
- status: Draft, Finalized
- reference_code: Generated code handed to interviewee

6.2.3 Building Assignment

Purpose: Track building assignments to field collectors

Fields:

- assignment_id: UUID
- building_id: Foreign key to Building
- field_collector_id: Foreign key to User
- assigned_date: Date
- transfer_status: Pending, Transferred, Failed
- units_for_revisit: Array of unit_ids selected for revisit

6.3 Entity Relationships

Key Relationships in the Data Model:

- Building → Property Units (1:N)
- Property Unit → Person-Unit Relations (1:N)
- Person → Person-Unit Relations (1:N)
- Person–Unit Relation → Evidence (1:N)
- Evidence → Documents (N:1)
- Property Unit → Occupancy Details (1:N)
- Building → Building Assignments (1:N)
- Building → Surveys (1:N)

7. Record ID Policy & Numbering System

Multi-Level Identifier Structure

The system uses three levels of identifiers to ensure comprehensive tracking:

Building Record ID (معرف سجل البناء)

A hierarchical 17-digit format for buildings:

Component	Digits	Description	Example
Governorate	2	محافظة	01 (Aleppo)
District	2	منطقة	01
Sub-district	2	ناحية	02
Community	3	تجمع عمراني	003
Neighborhood	3	حي	001
Building	5	رقم البناء	00001

Full Example: 01-01-02-003-001-00001

Property Unit ID (معرف الوحدة)

Component	Format	Description	Example
Building ID	17 digits	Building identifier	01-01-02-003-001-00001
Unit ID	3 digits	Unit within building	001

Full Example: 01-01-02-003-001-00001-001 (Unit 001 in Building 00001)

Claim ID (معرف المطالبة)

Component	Format	Description
Claim Prefix	CL	Fixed prefix
Year	4 digits	Year of submission
Sequential	6 digits	Sequential number

Full Example: CL-2025-000001

Legacy STDM Support

Field	Description	Usage
legacy_stdm_id	Original STDM identifier	Links to existing STDM database
migration_date	Date of import	Tracks when data was migrated
migration_status	Status flag	Validated/Pending/Conflicted

Informal Settlements Numbering

For unorganized areas (الأحياء الغير منظمة):

- Use special district codes (900-999)
- Assign sequential building numbers
- Maintain mapping table for area names

8. Controlled Vocabularies

Source Document: The following controlled vocabularies are based on the UN-Habitat required data specifications document. All vocabulary items listed below are subject to review and approval.

Version Control: Vocabularies follow semantic versioning (MAJOR.MINOR.PATCH) as described in Section 5.3 to ensure compatibility across system components.

8.1 Document Types (أنواع الوثائق)

A. Official/Legal Documents (وثائق رسمية/قانونية)

Code	Arabic Name	English Translation
TAPU_GREEN	صك ملكية (طبو أخضر)	Property Deed (Green TaPu)
PROPERTY_REG	بيان قيد عقاري	Property Registration Statement
TEMP_REG	بيان سجل مؤقت	Temporary Registration Statement
COURT_RULING	حكم قضائي	Court Ruling
POWER_ATTORNEY	سند توكيل خاص	Special Power of Attorney
IRR_POWER_ATTORNEY	وكالة غير قابلة للعزل	Irrevocable Power of Attorney
FINANCIAL_REG	بيان مالي	Financial Statement
REAL_ESTATE_TAXATION	إيصال ضريبة سنوية على العقار	Real estate taxation

B. Contracts/Civil Documents (عقود وكتابات مدنية)

Code	Arabic Name	English Translation
SALE_NOTARIZED	عقد بيع وشراء (موثق)	Notarized Sale Contract
SALE_INFORMAL	عقد بيع وشراء (غير موثق)	Informal Sale Contract
RENT_REGISTERED	عقد إيجار مثبت	Registered Rental Contract
RENT_INFORMAL	عقد إيجار غير مثبت	Informal Rental Contract

C. Proof of Residence (مستندات إثبات سكن)

Code	Arabic Name	English Translation
UTILITY_BILL	فاتورة مرافق عامة	Utility Bill
MUKHTAR_CERT	شهادة اختيارية	Mukhtar Certificate

F. Other Documents (أخرى)

Code	Arabic Name	English Translation
DECLARATION	إقرار وشطب	Declaration and Cancellation
INHERITANCE_LIMIT	حصر إرث	Inheritance Limitation Certificate

G. Verbal Statements (إفادات شفهية)

Code	Arabic Name	English Translation
CLAIMANT_STATEMENT	إفادة المدعي	Claimant Statement
WITNESS_STATEMENT	إفادة شاهد	Witness Statement
NEIGHBOR_TESTIMONY	شهادة جار	Neighbor Testimony
MUKHTAR_STATEMENT	إفادة المختار	Mukhtar Statement
COMMUNITY_AFFIRMATION	تأكيد مجتمعي	Community Affirmation

8.2 Occupancy Types (نوع الإشغال)

Code	Arabic	English
OWNER	مالك	Owner
TENANT	مستأجر	Tenant
GUEST	مستضاف	Guest
HEIR	ورثة	Heirs
OCCUPANT	شاغل	Occupant
NOT_SPECIFIED	غير معروف	Not Specified

9. Functional Requirements

9.1 Mobile Application Requirements

FR-M-Roles

Roles:

- Field Data Collector (Enumerator)
  - Works fully offline
  - It inputs: people – properties – units – visits – documents
  - Cannot see or modify the data of any other Enumerator

FR-M-1: Mobile App Screen Structure

The mobile application shall provide a screen structure and navigation flow that supports the field survey workflow.

Core Visit Screens (presented in guided sequence):

- Building Screen - Search and select the building
- Property Unit Screen - List and capture unit details
- Occupancy Details Screen - Capture occupancy information
- Persons Screen - Register persons and relation types
- Relation & Evidence Screen - Capture relations and evidence
- Final Review Screen - Consolidated summary and finalization

Global Navigation Screens:

- Home / Dashboard Screen
- Export / Sync Center Screen
- Settings / Configuration Screen
- Draft Surveys List - View and resume saved drafts

Acceptance Criteria:

- Screens navigable in sequence: Building → Property Unit → Household → Persons → Relation & Evidence → Final Review
- Finalization only possible after required stages completed
- All screens support Arabic (RTL)

FR-M-2: Data Capture

The app shall capture: Persons, Property units, Cases, Documents and attachments with comprehensive field coverage as specified in Section 6.

Acceptance Criteria:

- Complete data model implementation
- All required fields validated at entry
- Arabic text input supported throughout
- GPS coordinates captured when available

FR-M-3: Attachment Management

The app shall support multiple evidence documents per claim with photo/PDF attachments.

Acceptance Criteria:

- Multiple evidence documents per claim
- Maximum file size: 3 MB per image
- PDF multi-page file max 5MB
- Automatic compression before storage
- Deduplication by SHA-256 hash

FR-M-4: Field Validation

The app shall validate required fields and formats at data entry time.

Acceptance Criteria:

- National ID format validated (11 digits)
- Phone numbers normalized
- Dates cannot be in the future
- Required fields cannot be empty
- Year of birth validated as reasonable range

FR-M-5: Controlled Vocabularies

The app shall use downloadable, versioned vocabularies for all dropdown fields.

Acceptance Criteria:

- Vocabularies versioned (MAJOR.MINOR.PATCH)
- App blocks data entry if vocabulary incompatible
- Updates loaded from signed packages

FR-M-6: Unique Identifiers

Each record shall be assigned a globally unique UUID for internal technical purposes.

Important: Building/Unit codes captured separately according to UN-Habitat specifications.

FR-M-7: Audit Trail

The app shall maintain last_modified_utc, author_device, and operation flags.

FR-M-8: Data Transfer Mechanisms

The mobile app shall support TWO mechanisms for data transfer:

(a) Manual Export (.uhc Container): Export to signed SQLite container file

(b) Local-Network Synchronization: Bi-directional sync when on same LAN/Wi-Fi

FR-M-9: Data Package Manifest Structure

Manifest Components:

- package_id: UUID
- schema_version: Version of data schema
- created_utc: Timestamp
- device_id: Source device identifier
- app_version: Mobile app version
- vocab_versions: Vocabulary versions used
- form_schema_version: Form version

FR-M-10: Survey Reference Code Generation

The app shall generate a reference number/code at the end of each finalized survey that will be handed over to the interviewee by the field collector manually.

Acceptance Criteria:

- Reference code generated after completing/finalizing survey
- Code format includes date, collector code, and sequence number
- Code displayed for claimant to record manually
- Reference code maps to permanent Record ID after import

FR-M-11: Draft Survey Management

The app shall allow field collectors to save surveys as drafts and resume them later.

Acceptance Criteria:

- Surveys can be saved as Draft at any stage
- Draft surveys list shows all incomplete surveys with completion indicators
- Drafts can be filtered and searched
- Drafts can be resumed and completed
- Exit confirmation prompts to save as draft or delete

FR-M-12: Container Export

Users shall export data to signed SQLite container files (.uhc).

Acceptance Criteria:

- All data and attachments included
- Container includes manifest (FR-M-9)
- File signed with SHA-256 checksum
- Large datasets split into multiple containers with sequence numbers

9.2 Desktop Application Requirements

FR-D-1: Desktop Application Screen Structure

Core Screens:

- Import Wizard: Manual .uhc package import
- Case Search & Details: Advanced search with filters
- Data Entry Forms
- Claim Lifecycle Screen
- Documents/Attachments Preview
- Conflicts Review
- Data Quality Dashboards
- Maps
- Reports
- Admin Panel (Users/Roles/Vocab)
- Building Assignment Screen - Assign buildings to field teams
- Draft Office Surveys List
- Property Duplicate Review Queue
- Person Duplicate Review Queue

FR-D-2: Import Management

The desktop system shall ingest .uhc packages using manual and local-network methods.

Acceptance Criteria:

- Import Wizard supports drag-and-drop
- Package_id and checksum verified
- Duplicate imports ignored with notification
- Failed imports quarantined with error logs
- Invalid signature/hash packages quarantined
- Outdated vocabularies queued for manual update

FR-D-3: Validation & Verification

The system shall verify signatures, schemas, and vocabulary versions.

FR-D-4: Staging & Validation

The system shall stage data in a dedicated isolation layer and apply multi-level validation.

Staging detects anomalies and potential duplicates:

- Person matching: national_id, phone, Arabic name similarity
- Property matching: location codes, spatial proximity

FR-D-5: Person Matching (Detection)

The system shall match persons using a multi-attribute identity resolution algorithm.

Detection criteria: Persons flagged as potential duplicates when they share the same national_id value.

Detected duplicates are queued for human review in the Person Duplicate Review Queue (see FR-D-21).

FR-D-6: Property Matching (Detection)

The system shall match properties by location codes and spatial proximity.

Detection criteria: Properties flagged as potential duplicates when records share the same building_id or (building_id + unit_code) composite key.

Detected duplicates are queued for human review in the Property Duplicate Review Queue (see FR-D-20).

FR-D-7: Conflict Resolution

The system shall apply field-level conflict policies with human review option.

Resolution strategies: (a) merge as same, (b) keep as separate, (c) request field verification, (d) escalate to senior review.

FR-D-8: Record ID Generation

The system shall generate unique Record IDs on commit following 17-digit format.

FR-D-9: Attachment Deduplication

The system shall deduplicate attachments by SHA-256.

FR-D-10: Comprehensive Case Management

The desktop system shall create new claims, update existing claims, and perform quality control.

FR-D-11: Report Generation

The system shall generate Arabic PDF reports with QR codes.

FR-D-12: Dashboard Analytics

The system shall provide real-time dashboards for different audiences.

FR-D-13: API Integration Layer

The system shall provide RESTful APIs for system integration.

FR-D-14: Desktop Data Entry (Office Survey)

The desktop application shall provide data entry forms for in-office claim registration.

Office Survey workflow (per UC-004):

- Clerk has access to all buildings (not pre-filtered)
- Contact person can point to building on map or search by address
- Surveys saved to staging database directly
- Reference code generated for contact person
- Drafts can be saved and resumed later (UC-005)

FR-D-15: Local Network Sync Server

The desktop application shall act as a sync server on the local network.

FR-D-16: System Administration

Administrators shall manage users/roles, security policies, and controlled vocabularies.

Includes (per UC-009, UC-010, UC-011):

- User & Role Management: create/modify users, assign roles, manage permissions, enforce segregation-of-duties
- Vocabulary Management: import updates, add/edit/deprecate terms, set effective dates, export to tablets
- Security Settings: password policies, session timeout, lockout settings, access control policies

FR-D-17: Data Export & API Access

The system shall support standard exports (CSV/Excel/PDF/GeoJSON/RESTful APIs).

FR-D-18: Legacy STDM ID Support

The system shall support linking to legacy STDM IDs for data migration.

FR-D-19: Assign Buildings to Field Collectors

The system shall allow Data Managers to assign buildings to field collectors and transfer to tablets (per UC-012).

Acceptance Criteria:

- Open building assignment screen/map
- Search buildings by administrative hierarchy or locate on map
- Select building and review property units
- Select specific property units for revisit (if required)
- Move selected building to assigned buildings list
- Initiate transfer to LAN-connected tablet
- Handle transfer success (update status to 'Transferred', log event)
- Handle transfer failure (alert Data Manager, provide retry option)

FR-D-20: Resolve Duplicate Properties (Resolution)

The system shall provide a property duplicate review queue for resolving potential duplicates detected by FR-D-6 (per UC-007).

Acceptance Criteria:

- Display potential duplicate property groups
- Side-by-side comparison of property details
- Check location, geometry, and photos
- Select resolution strategy: merge, keep separate, request field verification
- Complex cases can be escalated to senior review queue
- Apply merge rules and update records
- Log all resolution decisions with justification

FR-D-21: Resolve Person Duplicates (Resolution)

The system shall provide a person duplicate review queue for resolving potential duplicates detected by FR-D-5 (per UC-008).

Acceptance Criteria:

- Display potential duplicate persons based on national_id
- Side-by-side comparison of person details

- Review ID documents with hash/verification metadata
- Select resolution strategy: merge, keep separate, request field verification
- Complex cases can be escalated to senior review queue
- Update linked relationships (ownership, household, claim foreign keys) after merge
- Log all resolution decisions

FR-D-22: Update Existing Claim

The system shall allow authorized users to update existing claims (per UC-006).

Acceptance Criteria:

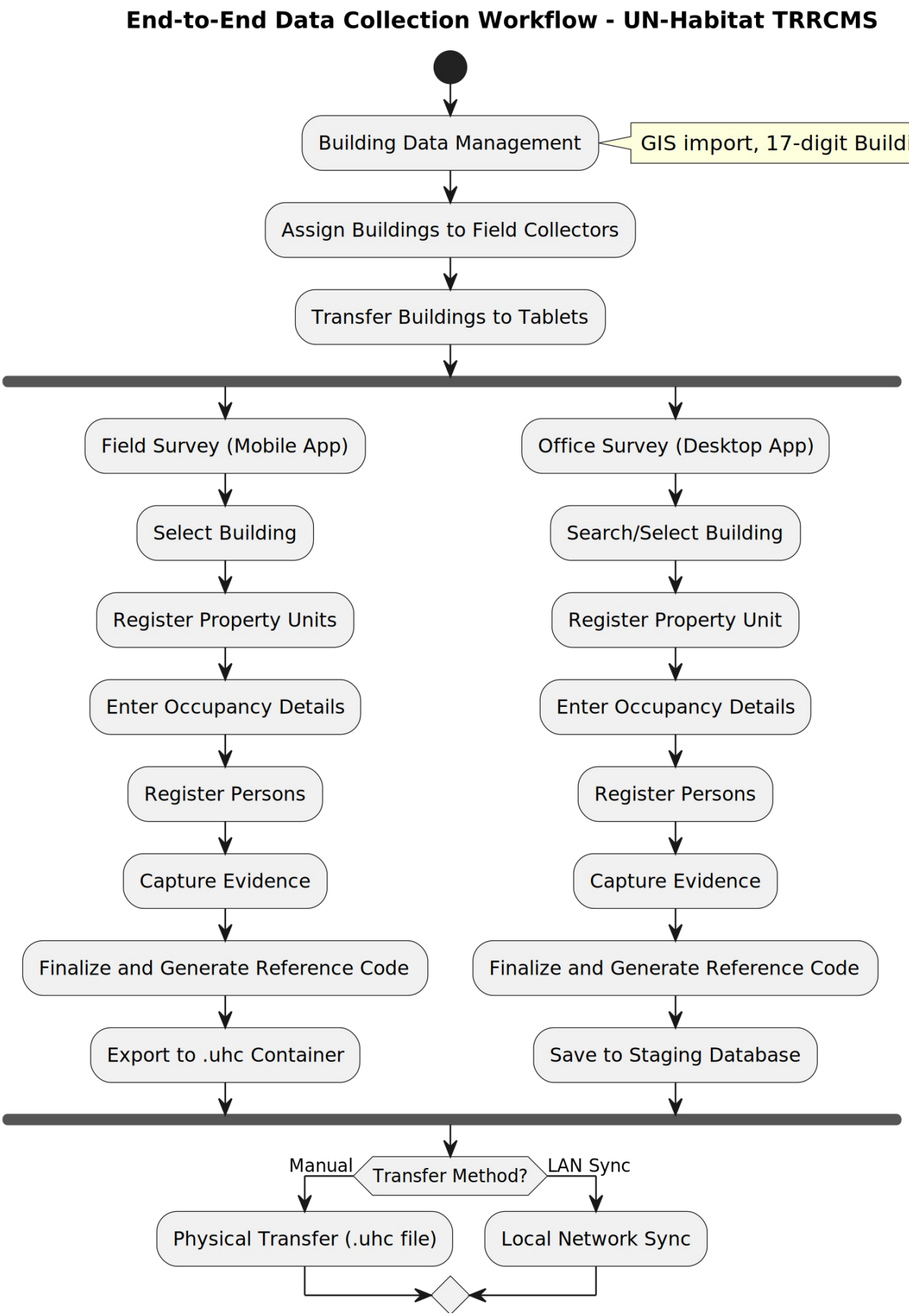
- Search for existing claims by Record ID, claimant name, date, or property unit
- View full claim details
- Apply updates: personal information corrections, property details, document additions/removals (versioning, not replacement), status changes
- Enter reason for modification (mandatory)
- Validate and save with audit trail
- Original record preserved in history
- Generate update confirmation

10. Workflows

This section provides visual representations of the key system workflows, illustrating the processes for data collection, import, validation, conflict resolution, and claim management.

WF-01: End-to-End Data Collection Workflow

This workflow illustrates the complete data collection process from planning through reporting, covering both field and office survey paths.

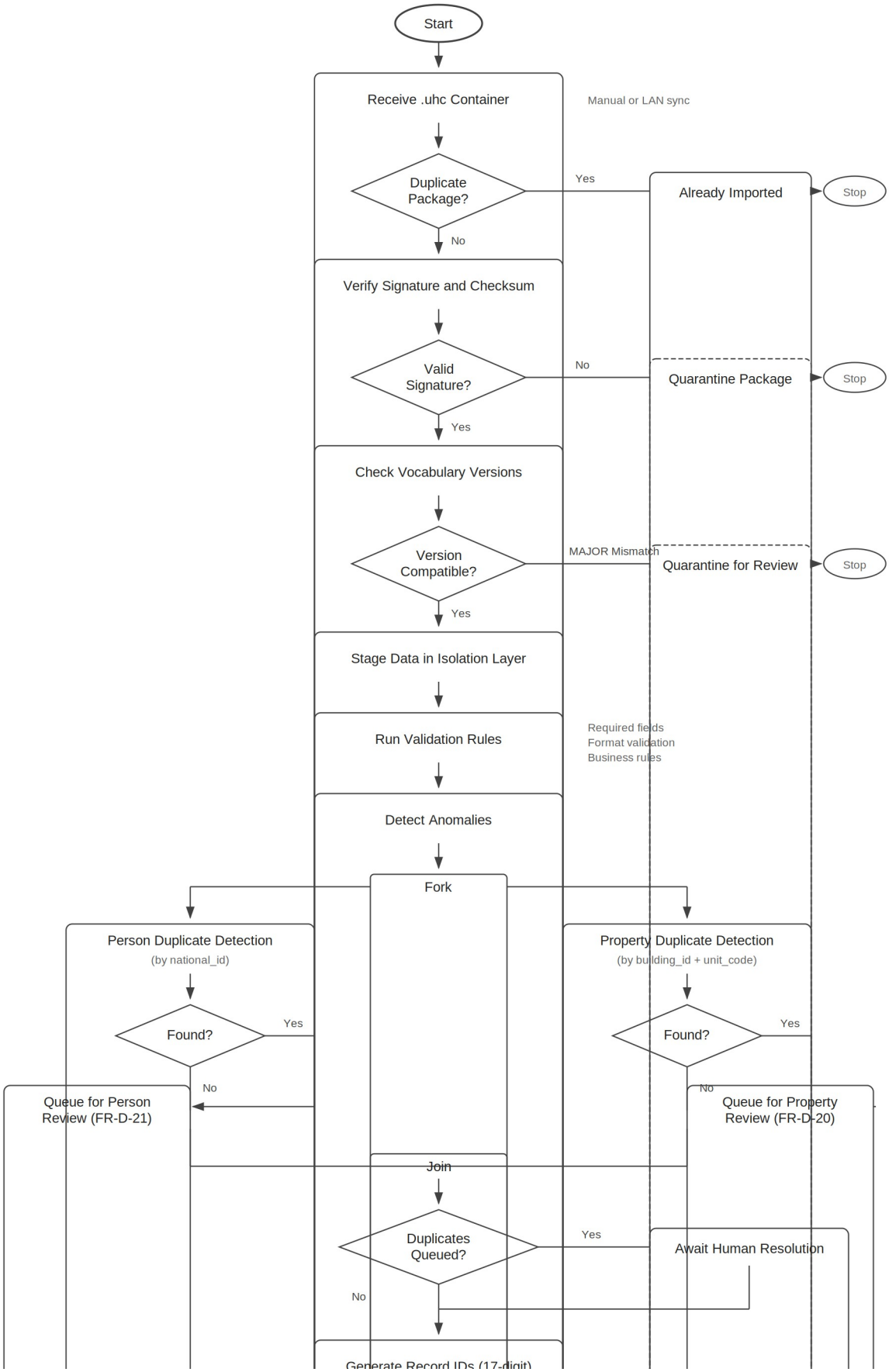

End-to-End Data Collection Workflow - UN-Habitat TRRCMS
Building Data Management
GIS import, 17-digit Building IDs
Assign Buildings to Field Collectors
Transfer Buildings to Tablets
Field Survey (Mobile App)
Select Building
Register Property Units
Enter Occupancy Details
Register Persons
Capture Evidence
Finalize and Generate Reference Code
Export to .uhc Container
Office Survey (Desktop App)
Search/Select Building
Register Property Unit
Enter Occupancy Details
Register Persons
Capture Evidence
Finalize and Generate Reference Code
Save to Staging Database
Transfer Method?
Manual
LAN Sync
Physical Transfer (.uhc file)
Local Network Sync

WF-02: Data Import and Validation Workflow

This workflow covers the import of .uhc containers, validation, duplicate detection, and commitment to the central database.

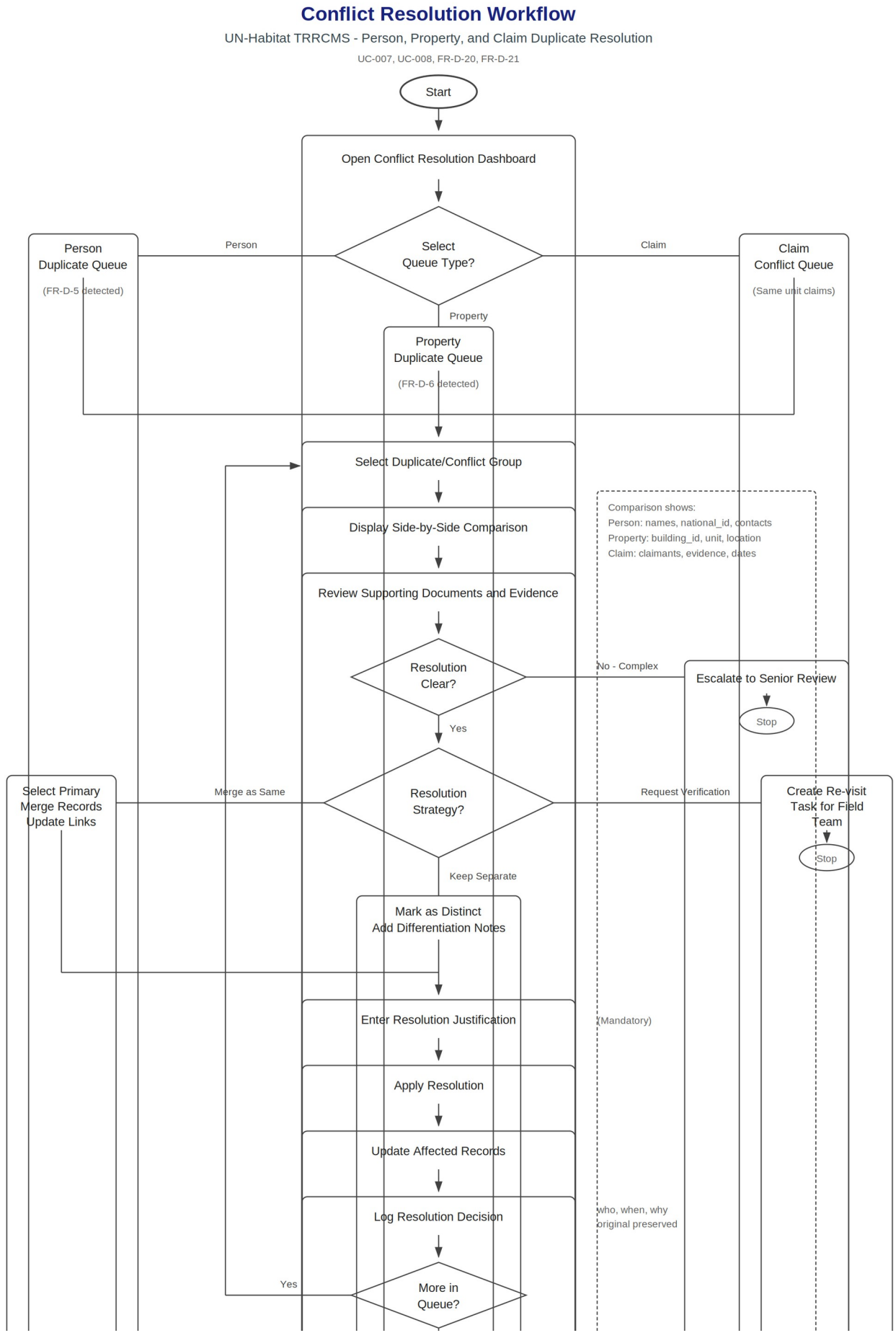
Data Import and Validation Workflow

UN-Habitat TRRCMS - FR-D-2, FR-D-3, FR-D-4, FR-D-5, FR-D-6



WF-03: Conflict Resolution Workflow

This workflow addresses the resolution of Person, Property, and Claim duplicates through the review queues.

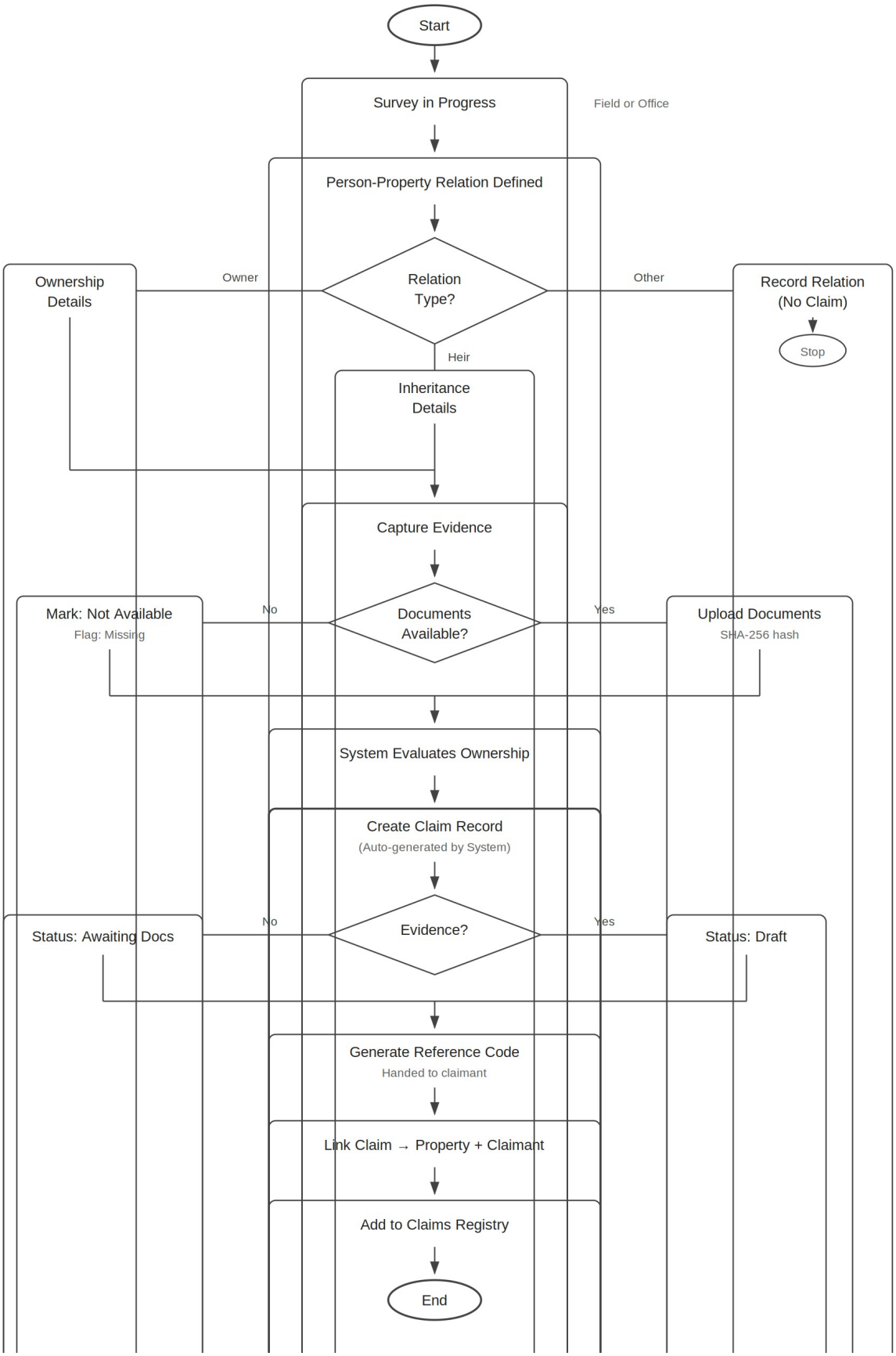

Conflict Resolution Workflow
UN-Habitat TRRCMS - Person, Property, and Claim Duplicate Resolution
UC-007, UC-008, FR-D-20, FR-D-21
Start
Open Conflict Resolution Dashboard
Select Queue Type?
Person
Property
Claim
Person Duplicate Queue
(FR-D-5 detected)
Property Duplicate Queue
(FR-D-6 detected)
Claim Conflict Queue
(Same unit claims)
Select Duplicate/Conflict Group
Display Side-by-Side Comparison
Comparison shows:
Person: names, national_id, contacts
Property: building_id, unit, location
Claim: claimants, evidence, dates
Review Supporting Documents and Evidence
Resolution Clear?
No - Complex
Escalate to Senior Review
Stop
Yes
Resolution Strategy?
Merge as Same
Request Verification
Select Primary Merge Records Update Links
Create Re-visit Task for Field Team
Stop
Keep Separate
Mark as Distinct Add Differentiation Notes
Enter Resolution Justification
(Mandatory)
Apply Resolution
Update Affected Records
Log Resolution Decision
who, when, why
original preserved
Yes
More in Queue?

WF-04: Claim Creation Workflow

This workflow shows how claims are automatically created during surveys when ownership relations are detected.

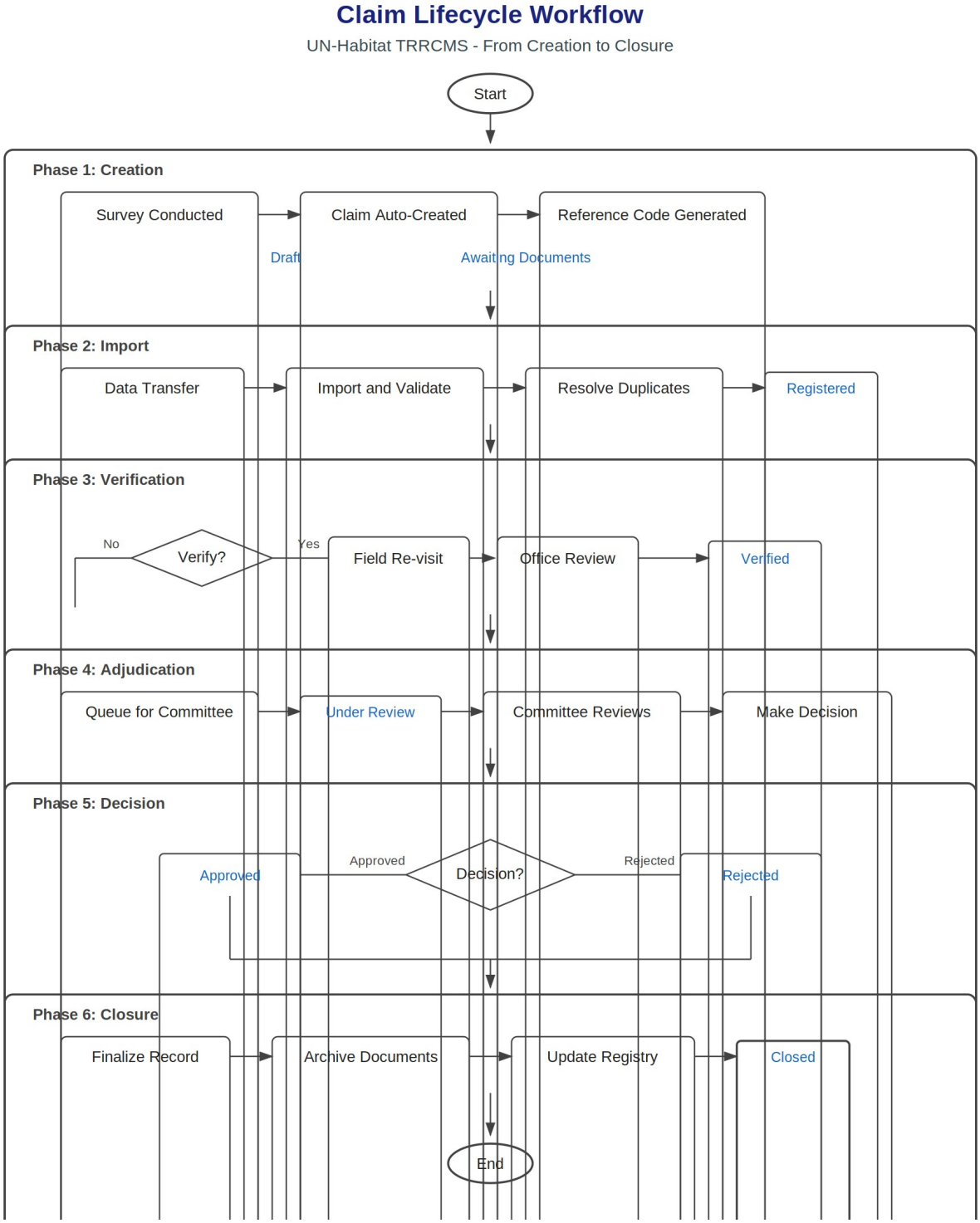
Claim Creation Workflow

UN-Habitat TRRCMS - Auto-generated during Survey (UC-001, UC-004)



WF-05: Claim Lifecycle Workflow

This workflow covers the complete claim lifecycle from creation through verification, adjudication, and closure.



11. Validation Rules

11.1 Data Entry Validation

Category	Field	Rule	Error Message
Person	First Name	Required, Arabic chars	الاسم الأول مطلوب
	Father Name	Required, Arabic chars	اسم الأب مطلوب
	Last Name	Required, Arabic chars	الكنية مطلوبة
	National ID	11 digits, unique	الرقم الوطني غير صحيح
	Mobile	Valid format	رقم الهاتف غير صحيح
Building	Governorate Code	2 digits (01-99)	رمز المحافظة غير صحيح
	District code	2 digits (01-99)	رمز المنطقة الإدارية
	Sub-District code	2 digits (01-99)	رمز الناحية الإدارية
	Community Code	3 digits (001-999)	رمز التجمع العمراني غير صحيح
	Neighborhood code	3 digits (000-999)	رمز الحي غير صحيح
	Building Number	5 digits (00001-99999)	رقم البناء غير صحيح
	Building Code	Must follow 17-digit pattern	رمز البناء غير مطابق للنمط المحدد
Unit	Unit Code	Must be unique within building	رمز الوحدة مكرر داخل البناء
	Unit Number	Required within building	رقم الوحدة مطلوب
Document	Type	From vocabulary	نوع الوثيقة غير معرف
	Issue Date	Not future	التاريخ غير صحيح
	Attachment	File exists, HASH present	الملف غير موجود
	File Size	<= configured max	حجم الملف يتجاوز الحد المسموح
	MIME Type	In allow-list	نوع الملف غير مسموح

11.2 Business Logic Validation

Building & Unit Coding Rules

- Building Code Pattern: Must follow 17-digit hierarchical format GG-DD-SS-CCC-NNN-BBBBB
- Building Code Uniqueness: No duplicate building codes allowed
- Unit Code Uniqueness: Unit codes must be unique within a building

Claim Structure Rules

A valid claim must always have:

1. At least one claimant (Person linked as claimant)
2. At least one unit (Property Unit linked to the claim)
3. At least one occupancy record
4. Document requirements (conditional per claim type)

Duplicate Prevention

- No duplicate National IDs in same import batch
- No duplicate building codes system-wide

- No duplicate Unit IDs within same building

12. Security & Audit

12.1 Security Requirements

Requirement	Implementation
Authentication	Username/password with 2FA option
Authorization	Role-based access control (RBAC)
Data Encryption	SQLCipher for .uhc files; central store encryption for attachments if required
Transmission	LAN TLS; supervised physical transfer for media
Audit Trail	All changes logged with timestamp, user, device, and action
Data Privacy	PII masked in logs
Hosting & Data Residency	On-premises deployment only; all data remains within controlled facility
Sensitive-Content Access	Role-based attachment access
Outputs Traceability	Signed and QR-stamped PDFs for all official outputs

12.2 Audit Trail Structure

Example audit log entry:

{ "event_id": "uuid", "timestamp": "2025-11-15T10:30:00Z", "user_id": "user123", "device_id": "TAB-001", "action": "UPDATE", "entity": "person", "entity_id": "uuid", "field": "mobile_number", "old_value": "xxx", "new_value": "yyy" }

12.3 Permissions Matrix

Action	Field Collector	Field Supervisor	Data Manager	Analyst	Admin
Create Data	✓	-	-	-	-
Export Container	✓	✓	-	-	-
Import Container	-	-	✓	-	✓
Resolve Conflicts	-	-	✓	-	✓
Generate Reports	-	✓	✓	✓	✓
View Sensitive Attachments	-	-	✓	-	✓
Manage Users	-	-	-	-	✓
Manage Vocabularies	-	-	-	-	✓

12.4 Legal Audit Trail Requirements

Complete Change History:

- All changes to any claim-related entity
- Who made each change (user ID, role, device)
- When the change was made (timestamp with timezone)
- What was changed (field-level with old and new values)

- Why the change was made (reason code or note)

Immutable Audit Log: Append-only, cryptographically signed, tampering detectable.

Reconstruction Capability: Point-in-time claim reconstruction, complete timeline view.

12.5 Retention and Archiving Strategy

- Active data in PostgreSQL with PostGIS
- Archived claims in separate archive tables
- All .uhc containers stored as immutable archive
- Attachments deduplicated and stored in content-addressable storage
- Backup: 3-2-1 rule (3 copies, 2 media types, 1 offsite)
- Legal Hold support for specific claims

13. Non-Functional Requirements

13.1 Performance

Requirement	Target	Measurement
Mobile app startup	<3 seconds	Cold start to home screen
Form save	<1 second	Submit to local storage
Container export	<30 seconds	1000 records with photos
Desktop import	<5 minutes	5000 records, 2GB attachments
Search response	<2 seconds	100K records
Report generation	<10 seconds	1000 records to PDF

13.2 Reliability

- Availability: 99.9% for desktop app during business hours
- Data Durability: No data loss after successful import
- Recovery: Automatic draft recovery on mobile app crash
- Idempotency: Duplicate imports cause no data corruption

13.3 Usability

- Language: Full Arabic support with RTL rendering
- Training: Maximum 2 hours for field collectors
- Error Messages: Clear, actionable, in Arabic
- Accessibility: Support for screen readers

13.4 Scalability

- Mobile: 100+ concurrent field devices
- Desktop: 10+ concurrent users
- Database: 1M+ total records
- Attachments: 100GB+ total storage

14. Reporting & Analytics

14.1 Dashboard Design

Reporting Dashboard (Executive): Total Claims, Claims by Status, Geographic Distribution, Monthly Trends

Data Validation Dashboard: Anomaly Detection, Validation Metrics, Building/Unit Coding Validity

Field Operations Dashboard: Daily Activity, Collector Performance, Coverage Maps

Compliance Dashboard: Audit Trail, Documentation Status, Regulatory Metrics

GIS Dashboard: Spatial Coverage Map, Damage Assessment Overlay, QGIS Export

UN-Habitat Programme Management Dashboard: Programme KPIs, Field Operations Efficiency, Donor Reporting

14.2 Data Extraction Features

- Export Formats: CSV, Excel, PDF, GeoJSON, RESTful APIs
- Value Codes: Include both codes and display names
- Custom Queries: Build custom filters with field selection
- Scheduled Exports: Automated daily/weekly exports
- Incremental Exports: Export only changes since last export

14.3 Standard Reports

Report Name	Description	Format
Daily Collection Summary	Records collected per day/collector	PDF, Excel
Property Status Report	Properties by damage/occupancy status	PDF, Excel
Document Completeness	Missing documents by case	Excel
Spatial Distribution	Cases by geographic area	GeoJSON, PDF
Conflict Resolution Log	Manual reviews and decisions	PDF
Import Summary	Success/failure per import batch	HTML, PDF

14.4 Export Formats

Format	Use Case	Contents
CSV	Data analysis	Flat structure, all fields
Excel	Management review	Multiple sheets, formatted
GeoJSON	GIS analysis	Spatial data with attributes
PDF	Official documents	Formatted, signed, with QR
XML	System integration	Structured, schema-validated

14.5 QGIS Interoperability

REST API Endpoints: /api/geo/buildings, /api/geo/units, /api/geo/claims, /api/geo/damage, /api/geo/occupancy

Mapping Services: WFS, WMS, PostGIS spatial indexes, CRS support (EPSG:4326)

14.6 Power BI Integration

Secure Read-Only Analytics API with OAuth2, refresh scheduling, optimized query performance.

14.7 STDM Integration

FSD Entity	STDM Entity	Mapping Notes
FSD.Building	STDM.SpatialUnit	Building geometry and attributes
FSD.Owner	STDM.Party	Owner persons with rights
FSD.Claimant	STDM.Party	Claimant persons with claims
FSD.Documents	STDM.Evidence	Supporting documentation

15. Acceptance Criteria

15.1 Functional Acceptance

Criteria	Test Method	Pass Condition
Data Collection	Field test	100 records captured offline
Container Export	Verification	Valid .uhc with checksum
Import Process	End-to-end test	5000 records imported
Duplicate Detection	Test dataset	95% accuracy
Record ID Generation	Uniqueness test	No collisions in 10K records
Arabic Support	UI review	All text renders correctly
Report Generation	Sample data	All reports generate <10s

15.2 Performance Acceptance

Criteria	Target	Test Method
Mobile Performance	<1s saves	Timed operations
Import Speed	>1000 records/min	Batch import test
Query Response	<2s for 100K	Load testing
Concurrent Users	10+ desktop	Stress testing

15.3 Definition of Done

- All functional requirements implemented
- All use cases executable end-to-end
- Arabic UI completely functional
- Validation rules enforced
- Audit trail complete
- Reports generating correctly
- Performance targets met
- User acceptance testing passed
- Documentation complete
- Training materials ready

Appendices

Appendix A: Glossary (مسرد المصطلحات)

English	Arabic	Description
Record ID	رقم المحضر	Unique 17-digit identifier
Container	حاوية	.uhc export file
Vocabulary	مفردات	Controlled dropdown values
Staging	التدريج	Temporary validation area
Deduplication	إزالة التكرار	Removing duplicate records
Idempotency	عدم التأثر	Safe repeat operations

Appendix B: Administrative Domains

Detailed hierarchical codes for Aleppo governorate administrative divisions to be provided by local authorities.

Appendix C: Error Codes

Code	Category	Message (AR)	Resolution
E001	Validation	حقل مطلوب	Complete required field
E002	Format	تنسيق غير صحيح	Check format requirements
E003	Duplicate	سجل مكرر	Review existing record
E004	Network	لا يوجد اتصال	Work offline
E005	Storage	مساحة غير كافية	Free device space

Appendix D: API Specifications

D.1 API Endpoints

Endpoint	Method	Description	Authentication
/api/v1/claims	POST	Submit new claim	OAuth 2.0
/api/v1/claims/{id}	GET	Retrieve claim details	OAuth 2.0
/api/v1/claims/{id}	PUT	Update existing claim	OAuth 2.0
/api/v1/claims/search	POST	Search claims	OAuth 2.0
/api/v1/documents/{id}	GET	Retrieve document	OAuth 2.0
/api/v1/statistics	GET	Get statistics	API Key
/api/v1/vocabularies	GET	Get controlled vocabularies	Public
/api/v1/export	POST	Bulk data export	OAuth 2.0

D.2 Webhook Events

- claim.created - New claim registered
- claim.updated - Claim information updated

- claim.approved - Claim approved
- claim.rejected - Claim rejected
- document.verified - Document verification completed

End of Document