# Use Cases

## 1. Use Cases

Interactions between users and the system, described in a narrative or structured manner.

### Use Case 1: Register Problem (Online)

* **Identifier:** UC001 (Related to RF01, US001)
* **Name:** Register Problem (Online)
* **Main Actor:** Consumer (Primary Actor)
* **Description:** This Use Case describes the process by which a Consumer of the client company (e.g., Natwest) logs a new complaint or issue via the web or mobile application access points.
* **Pre-Condition:** Consumer has access to the web or mobile application. System is operational (24/7 availability requirement, RNF01).
* **Main Flow:**
  1. The Consumer navigates to the 'Report a Problem' interface.
  2. The System displays the required fields for problem logging (e.g., type, description, contact details).
  3. The Consumer inputs the necessary details and submits the report.
  4. The System validates the submitted data for completeness and correctness.
  5. The System creates a new problem record in the database, assigns a unique tracking ID, and sets the initial status to 'Logged'.
  6. The System sends a confirmation message, including the tracking ID, to the Consumer.
  7. Use Case ends successfully (Success Guarantee: Problem record exists; tracking ID provided).
* **Alternative Flows / Exceptions:**
  + **A1: Consumer Requests Telephone Support:** At Step 1, the Consumer selects the option to call the Help Desk instead (UC009). The system displays the relevant company’s contact number (RN02).
  + **E1: Invalid/Missing Data:** At Step 4, the System identifies missing required information. The System displays an error message indicating the fields that need correction. Process returns to Step 2.
  + **E2: System Failure:** At Step 5, the system fails to create the record due to a temporary error. The System displays a minimal guarantee message ensuring that data input integrity is maintained (Minimal Guarantee: No data is lost; Consumer is informed that logging failed and should retry). Use Case terminates unsuccessfully.

### Use Case 2: Assign Problem to Specialist

* **Identifier:** UC002 (Related to RF02, US002)
* **Name:** Assign Problem to Specialist
* **Main Actor:** Help Desk Agent (Primary Actor)
* **Description:** This Use Case handles the process where a Help Desk Agent, after initial triage, determines that a problem requires the expertise of a specialized Support Person.
* **Pre-Condition:** Help Desk Agent is logged in. A problem record exists and requires specialized resolution.
* **Main Flow:**
  1. The Help Desk Agent reviews the problem details.
  2. The Agent selects the "Assign Specialist" function.
  3. The System displays a list of available Support Persons categorized by specialization.
  4. The Agent selects the most appropriate Support Person from the list.
  5. The System validates the assignment and updates the problem status to 'Assigned'.
  6. The System sends a notification to the selected Support Person regarding the new assignment.
  7. Use Case ends successfully (Success Guarantee: Problem status is 'Assigned'; Support Person is notified).
* **Alternative Flows / Exceptions:**
  + **A1: Selected Specialist Unavailable:** At Step 4, the System informs the Agent that the selected Support Person is unavailable or over capacity. The System prompts the Agent to select another available specialist. Process returns to Step 3.
  + **E1: Assignment System Error:** At Step 5, the assignment fails due to a database error. The System displays an error message and logs the attempt. Use Case terminates unsuccessfully (Minimal Guarantee: Problem remains unassigned and status unchanged).

### Use Case 3: Check Problem Resolution Status

* **Identifier:** UC003 (Related to RF04, US003)
* **Name:** Check Problem Resolution Status
* **Main Actor:** Consumer (Primary Actor)
* **Description:** This Use Case allows the Consumer to track the status of a previously logged problem via online channels.
* **Pre-Condition:** Problem has been logged and the Consumer has the tracking ID (from UC001).
* **Trigger:** Consumer enters the tracking ID into the status check field.
* **Main Flow:**
  1. The Consumer inputs the tracking ID into the designated online status field.
  2. The System validates the tracking ID format and authenticity.
  3. The System queries the problem database using the ID.
  4. The System retrieves the latest status (e.g., 'In Progress', 'Resolved').
  5. The System displays the current status and last update date to the Consumer.
  6. Use Case ends successfully (Success Guarantee: Latest status is displayed).
* **Alternative Flows / Exceptions:**
  + **A1: Status Update via Message/Email:** At Step 4, if the consumer is not logged in, the System automatically sends an email or message (depending on preference) with the current status, fulfilling the multi-channel requirement.
  + **E1: Invalid Tracking ID:** At Step 2, the System determines the ID is invalid or not found. The System displays an error message informing the Consumer to check the ID and retry. Use Case terminates unsuccessfully.

### Use Case 4: Resolve Problem (Specialist)

* **Identifier:** UC004 (Related to RF05, US004)
* **Name:** Resolve Problem (Specialist)
* **Main Actor:** Support Person (Primary Actor)
* **Description:** This Use Case covers the actions of a specialized Support Person who resolves an assigned problem and formally updates the system with the solution notes.
* **Pre-Condition:** Problem status is 'Assigned' (UC002). Support Person is logged in.
* **Main Flow:**
  1. The Support Person accesses the assigned problem record.
  2. The Support Person executes the necessary steps to resolve the problem externally (e.g., fixing a technical issue).
  3. The Support Person inputs the formal resolution notes and selects the status 'Resolved'.
  4. The System validates the notes and updates the problem record.
  5. The System notifies the originating Help Desk Agent and triggers a notification to the Consumer informing them the problem is resolved.
  6. Use Case ends successfully (Success Guarantee: Problem status is 'Resolved'; Resolution notes are logged).
* **Alternative Flows / Exceptions:**
  + **A1: Resolution Requires Customer Input:** At Step 3, the Support Person notes that a final action is required by the Consumer. Status is set to 'Resolution Pending Customer'. Flow continues to Step 5 (notifying Consumer).
  + **E1: Resolution Not Possible:** At Step 3, the Support Person determines the problem cannot be solved or requires escalation. Status is set to 'Escalated'. The System notifies the Manager. Use Case terminates unsuccessfully (Minimal Guarantee: All attempts and reasons for failure are logged).

### Use Case 5: Close Problem

* **Identifier:** UC005 (Related to RF06, US005)
* **Name:** Close Problem
* **Main Actor:** Help Desk Agent (Primary Actor)
* **Description:** This Use Case finalizes the problem lifecycle by formally closing the record, which is strictly dependent on the Consumer confirming satisfaction (Business Rule RN03).
* **Pre-Condition:** Problem status is 'Resolved' (from UC004 or UC007). Consumer satisfaction is confirmed (online or via phone).
* **Trigger:** Consumer provides explicit satisfaction confirmation.
* **Main Flow:**
  1. The Help Desk Agent confirms that the Consumer is satisfied with the resolution (RF06).
  2. The Agent selects the "Close Problem" function on the problem record.
  3. The System performs a final check for mandatory closure notes.
  4. The System changes the problem status to 'Closed' and archives the record.
  5. Use Case ends successfully (Success Guarantee: Problem status is 'Closed'; lifecycle ends).
* **Alternative Flows / Exceptions:**
  + **E1: Consumer Reports Dissatisfaction:** At Step 1, the Consumer expresses that the problem is not solved or a new issue has arisen. The Agent stops the closure process. The Agent changes the status back to 'Re-opened' and returns the problem for re-assignment (UC002). Use Case terminates unsuccessfully (Minimal Guarantee: Problem integrity is maintained; problem is not closed prematurely).

### Use Case 6: Monitor Resolution Performance

* **Identifier:** UC008 (Related to RF07, US006)
* **Name:** Monitor Resolution Performance
* **Main Actor:** Help Desk Manager (Primary Actor)
* **Description:** This Use Case allows the Help Desk Manager to generate reports on key performance indicators (KPIs) like resolution time and staff performance to facilitate data-driven decisions.
* **Pre-Condition:** Manager is logged in. Historical problem data exists in the system.
* **Trigger:** Manager initiates the report generation process.
* **Main Flow:**
  1. The Help Desk Manager selects the desired monitoring criteria (e.g., time period, staff type: Agents or Support Persons).
  2. The System processes the request by querying historical problem records.
  3. The System calculates the required metrics (e.g., average resolution time, individual staff performance).
  4. The System generates and displays the performance report in a graphical and/or tabular format.
  5. Use Case ends successfully (Success Guarantee: Performance metrics are displayed).
* **Alternative Flows / Exceptions:**
  + **A1: Export Report:** At Step 4, the Manager selects the option to export the report data (e.g., CSV, PDF). The System generates and provides the file for download.
  + **E1: Insufficient Data:** At Step 3, the System determines that the selected criteria or time period contains insufficient data for meaningful calculation. The System displays a warning message. Use Case terminates unsuccessfully (Minimal Guarantee: No incorrect data is presented).

### Use Case 7: Resolve Problem (Agent)

* **Identifier:** UC007 (Related to RF03, US008)
* **Name:** Resolve Problem (Agent)
* **Main Actor:** Help Desk Agent (Primary Actor)
* **Description:** This Use Case covers the streamlined process where a Help Desk Agent resolves a common problem immediately without needing to assign it to a specialist.
* **Pre-Condition:** Problem status is 'Logged' (UC001). Agent is logged in.
* **Main Flow:**
  1. The Help Desk Agent accesses the logged problem record.
  2. The Agent applies an immediate, known solution (e.g., guiding the Consumer through a simple setting change).
  3. The Agent inputs brief resolution notes and changes the status to 'Resolved by Agent'.
  4. The System validates the notes and updates the record.
  5. The System notifies the Consumer that the problem is resolved and awaits confirmation (UC005).
  6. Use Case ends successfully (Success Guarantee: Problem status is 'Resolved by Agent'; Consumer notified).
* **Alternative Flows / Exceptions:**
  + **A1: Requires Specialist Intervention:** At Step 2, the Agent determines the problem is more complex than anticipated. The Agent initiates UC002 (Assign Problem to Specialist).
  + **E1: Consumer Disconnects:** At Step 2, the Consumer is disconnected or fails to respond. The Agent changes the status to 'Resolution Attempted - Awaiting Contact'. Use Case terminates unsuccessfully.

### Use Case 8: Onboard New Organization

* **Identifier:** UC006 (Related to RF08, US007)
* **Name:** Onboard New Organization
* **Main Actor:** System Administrator (Primary Actor)
* **Description:** This Use Case details the process for the System Administrator to set up a new client company (e.g., Natwest) within the multi-tenant architecture.
* **Pre-Condition:** Administrator is logged in. Necessary organization details are available.
* **Main Flow:**
  1. The System Administrator selects the 'Onboard New Organization' function.
  2. The System displays configuration fields (e.g., Organization Name, Data Partition ID, Initial User Roles).
  3. The Administrator inputs and confirms the required details.
  4. The System executes the multi-tenant setup process, creating a new, isolated data partition (RNF04).
  5. The System confirms successful onboarding and generates initial administrator credentials for the new organization.
  6. Use Case ends successfully (Success Guarantee: New organization is set up with isolated data access).
* **Alternative Flows / Exceptions:**
  + **E1: Data Isolation Failure:** At Step 4, the System identifies a potential security conflict (e.g., non-unique Data Partition ID). The System stops the process and displays an error warning regarding the integrity requirement (RNF04). Use Case terminates unsuccessfully.

### Use Case 9: Log Problem via Telephone

* **Identifier:** UC009
* **Name:** Log Problem via Telephone
* **Main Actor:** Help Desk Agent (Primary Actor, logs problem), Consumer (Secondary Actor, requests logging)
* **Description:** This Use Case describes the process where a Consumer calls the Help Desk during operational hours (RN02), and the Agent manually logs the problem into the CMS on their behalf.
* **Pre-Condition:** Consumer has reached the Help Desk Agent. Telephone services are operating within standardized hours (RN02).
* **Trigger:** Help Desk Agent receives a call from the Consumer requesting to log a problem.
* **Main Flow:**
  1. The Help Desk Agent receives the call and verifies the Consumer's identity.
  2. The Agent opens a new problem registration form in the CMS interface.
  3. The Agent gathers the required problem details from the Consumer.
  4. The Agent submits the problem details to the System.
  5. The System processes the submission, creates a problem record, and generates a tracking ID.
  6. The Agent verbally relays the tracking ID and confirms the next steps to the Consumer.
  7. Use Case ends successfully (Success Guarantee: Problem record exists; tracking ID provided to Consumer).
* **Alternative Flows / Exceptions:**
  + **E1: Call Dropped/Verification Failed:** At Step 1, identity verification fails or the call is dropped before logging is complete. The Agent sets the status as 'Incomplete - Awaiting Contact'. Use Case terminates unsuccessfully (Minimal Guarantee: Basic consumer details are recorded).

## General Observations

* Use Cases UC001 (Online), UC009 (Telephone), and UC003 (Status Check) directly address the multi-channel access points (web, mobile, telephone) required by the system specification.
* All Use Cases involving system completion (UC005, UC004, UC007) must adhere to the Minimal Guarantee Post-condition ensuring non-functional requirements (like data integrity) are preserved even on failure.