Minutes of Meeting

Date and Time	08 October 2025 09:00 PST	**Meeting type**	Zoom
Organiser	Mr. Rupesh	**Client**	Citywide

Attendees (Internal)
• Rupesh • Jaspreet • Ravinder • Rahul • Ajay • Gurpreet • Kapil • Pankaj
Attendees (Client Side)
• Tom, Teresa, Randy

Agenda

• **Discussions on the following:**
  ○ Forms Linking Implementations
  ○ Communication Gaps & Repeated Discussions
  ○ Analytics Demonstration

The following things are discussed:

1. **Forms Linking Implementations:**
    a. **Form Viewing and Demonstration:**
        i. **Discussion:**
            1. Ajay demonstrated how forms linked to reports can be viewed under “My Reports.”
            2. The system allows multiple forms (e.g., Parking Citation, Field Interview, Incident Report) to be associated with a single report.
            3. From the report view page, users can directly access and open linked forms via the “View” button.
        ii. **Decision:**
            1. The demonstration was approved as functioning correctly.
            2. No changes required in the current viewing flow.
    b. **Handling Linked Forms During Report Deletion or Activity Code Changes**
        i. **Discussion:**
            1. Jaspreet raised scenarios related to **deletion or modification of reports** and how that should affect linked forms:
                a. If a report is deleted, should all forms linked to it also be deleted?
                b. If the **activity code** of a report is updated, should linked forms change or remain?
            2. Randy and Tom brought up the field scenario:
                a. A call might come in with one activity code (e.g., noise complaint) but later, upon arrival, the officer may realize it’s a different case (e.g., murder).
            3. Teresa clarified that the **original call code** should remain as-is, representing what the resident reported.
                a. The report may have a different **activity code** to represent actual findings.
                b. The call and report therefore serve different purposes and should not overwrite each other.
            4. Jaspreet questioned how this change would affect form linking.
            5. Teresa confirmed that once a form is filled out, it becomes a **standalone entity** — changing the activity code later should not delete or unlink the existing form.
        ii. **Decision:**
            1. **Forms once created remain independent**, even if the activity code of the report changes.
            2. The **call activity code** should remain locked to represent the caller’s input.
            3. **Reports can have an updated activity code**, which may trigger new form requirements without affecting old forms.
    c. **Clarification on “Yes/No” Form Redirection Flow**
        i. **Discussion:**
            1. The current popup text says “OK/Cancel” when an activity code requiring a form is selected.
            2. Tom requested to change this to **“Yes/No”** for clarity:
                a. “Field Interview form is required for this activity. Do

you want to go to the form?"

3. Jaspreet and Teresa confirmed this change is needed for better user understanding.
4. Randy and Teresa suggested that selecting "Yes" should immediately **redirect users to the respective form**, not after saving the report.

ii. **Decision:**

1. The popup will display **"Yes/No"** instead of "OK/Cancel."
2. On selecting **"Yes"**, the system should **auto-save the report** and then **redirect** to the linked form immediately.

d. Separation of Two Functional Flows: Report vs Call

i. **Discussion:**

1. Teresa emphasized that the confusion arises because **two flows are being mixed**:
    a. **Report-based flow:** Activity code triggers form suggestions when creating a standalone report.
    b. **Call-based flow:** When a report is created from a call, the activity code and site are inherited and often locked.
2. For **report-only flow:**
    a. The activity code will determine whether a form is needed.
    b. If "Yes" is selected, the user goes directly to the form without saving the report first.
3. For **call-based flow:**
    a. The call details remain constant (activity code, site, etc.).
    b. If the officer changes the activity code (e.g., discovers a more serious event), it can prompt a new form requirement.
    c. In this case, the system will **auto-save** the report before opening the form so that report and form remain linked.

ii. **Decision:**

1. Clearly separate **Report flow** and **Call flow** logic in the backend.
2. **Report flow:** No mandatory linking between reports and forms.
3. **Call flow:** Linking remains based on the selected activity code and associated form mapping.

e. Locking and Updating Activity Codes

i. **Discussion:**

1. Tom proposed that the **original call activity code** must be **locked** and not editable.
2. A new field — **"Did the activity code remain the same?" (Yes/No)** — should be added.
    a. If **"Yes"**, no change required.
    b. If **"No"**, user can select a new secondary activity code.
    c. The new activity code can then trigger form suggestions as per configuration.
3. This approach solves the issue of changing call codes without losing historical accuracy.

ii. **Decision:**

1. Original **call activity code** → locked.
2. Add a **Yes/No confirmation field** for activity code changes.
3. If changed, the system will trigger the appropriate form based on the new activity code.

f. System Behavior for Form Prompt and Report Linking

i. Discussion:

1. Tom explained the expected flow:
    a. When a user selects "Yes" to a required form prompt, the system should automatically:
        i. Save the report and generate a report ID.
        ii. Redirect to the corresponding form.
        iii. Maintain a link between the new form and saved report.
    b. Teresa asked whether this logic applies both to calls and independent reports.
    c. Tom confirmed:
        i. **Yes** for both cases, but in **calls**, the call ID ensures linkage.
        ii. In standalone reports, the report ID acts as the reference.

ii. Decision:

1. Implement **auto-save + redirect** functionality when "Yes" is selected on form prompt.
2. Maintain correct linking (Report ID or Call ID as applicable).
3. No separate manual save required before form redirection.

g. Additional Clarifications

i. **Randy** asked whether officers still need to fill out disposition fields when redirected to a form.
    1. Teresa confirmed that forms already contain disposition fields where necessary.

ii. **Tom** emphasized the functional consistency between call-based and standalone report workflows.

2. Communication Gaps & Repeated Discussions

a. **Tom** highlighted that the same topic of **linking reports, incidents, and activity codes** has been discussed for several weeks without final resolution.
b. The repeated discussions stem from a **lack of clear technical communication** between the client (Teresa) and the development team (Ditstek).
c. He pointed out that when Ditstek mentioned needing a report number before linking, it became evident that both sides were not fully aligned on technical requirements.
d. Tom expressed that both teams share responsibility —
    i. Teresa might not have fully explained their intended functionality.
    ii. Ditstek may not have clearly communicated the dependencies and workflow requirements for implementation.
e. He suggested improving the **communication workflow** to prevent such circular discussions and accelerate progress.

f. Proposal for Process Improvement

i. Kuldeep proposed a structured approach to avoid ambiguity in future development cycles:
    1. Creation of Workflow Documentation:
        a. After every call or discussion, Ditstek will create a **workflow understanding document** summarizing:

i. The discussed points.
ii. Agreed logic and dependencies.
iii. Implementation notes.

b. This document will act as a reference for both teams before development begins.

2. **Wireframing Before Development:**
   a. A **UI wireframe** should be created prior to any backend development.
   b. The wireframe will visually demonstrate:
      i. Step-by-step user flow (e.g., form selection → report linking → submission).
      ii. UI transitions and screen interactions.
      iii. Expected data movement without backend logic.
   c. This visual clarity will help both teams align expectations and validate the flow before backend work starts.
3. **Review & Sign-Off:**
   a. Once wireframes and documentation are reviewed and finalized by both parties, only then the development team will proceed with backend implementation.
   b. This approach ensures the workflow is **finalized and approved**, reducing rework and confusion.

3. **Analytics Demonstration**
   a. **Analytics Demonstration**
      i. **Presented by:** Gurpreet
         1. The demo covered the **Company-level analytics tab**.
         2. The first graph displayed was **"Service Type Based Site Distribution"** comparing **Current Period** vs **Previous Period**.
         3. Clicking on graph segments redirects users to the **Site screen**, where relevant filters (date range, type, etc.) are automatically applied.
         4. The linking between analytics and respective tabs (e.g., site view) is functional.
      ii. **Feedback from Tom:**
         1. Confirmed that clicking on colored segments (e.g., blue) should display lists filtered by the selected type (e.g., Mobile sites only).
         2. Requested consistency across analytics — e.g., when selecting "Yearly," all displayed data should align with the yearly period, not mixed (monthly vs yearly).
   b. **Clarification on Period Comparisons**
      i. **Discussion Points:**
         1. Gurpreet mentioned that for **Yearly**, a total of 365 days is considered; for **Monthly**, a fixed 30-day comparison is used.
         2. Tom clarified that **comparisons must align with calendar periods**, not fixed day counts.
      ii. **Key Guidelines from Tom:**
         1. **Daily:**
            a. Users should be able to pick any range (e.g., 2 days, 62 days) and compare it with the **exact same previous range**.

2. **Monthly:**
    a. When the user selects a month (e.g., September), the system must automatically compare with the **previous calendar month (August)**, even if day counts differ (30 vs 31 days).
3. **Quarterly / Yearly:**
    a. Must use **calendar quarters** and **calendar years**.
    b. Leap years or day-count variations should not impact period logic.
4. **Custom Range:**
    a. Should behave as the current "Daily" setup — compare custom date range with an identical previous range.

iii. **Tom's Example Reference:**

1. He demonstrated using QuickBooks logic, where reports are compared across **calendar-defined periods**.
2. Stressed that day-based comparisons (e.g., fixed 30 days) lead to data inaccuracies.

c. **Advanced Reporting Expectations**

i. **Tom's Requirements for Next Updates:**

1. The system should not only show **statistical differences** between periods but also include **reasons ("Why")** for variations.

ii. Example causes to be captured:

1. Gain/Loss of clients
2. Holidays or operational downtime
3. Difference in days (e.g., 31 vs 30 days)
4. Other contextual business factors

iii. **Ultimate Goal:**

1. "If the number has changed, I want to know why." – Tom

d. **Discussion on Implementation Priorities**

i. **Jaspreet's Input:**

1. Suggested deploying the **stable analytics components** first (the working filters and linking).
2. The advanced period comparison and "why analysis" can follow in subsequent updates.

ii. **Tom's Response:**

1. Declined partial deployment until **monthly reports show accurate calendar-based comparisons**.
2. Emphasized correctness over speed of release.
3. Agreed that once period logic and filters align with calendar structure, analytics can go live.

e. **Export Functionality Discussion**

i. **Overview:**

1. Gurpreet showcased the newly added **Export button** that exports data currently visible under applied filters.
2. Tom requested that the export output should **replicate the exact on-screen view** — not just raw data.

ii. **Key Discussion Points:**

1. Tom emphasized that **export as PDF** must look **identical to the analytics screen view**, including graphs, layout, and filters.
2. The export should function as a **snapshot**

(screenshot-style export) of the current display.
3. The export options should include:
    a. PNG (visual snapshot)
    b. CSV (raw data)
    c. PDF (formatted report)

iii. Technical Clarification:

1. Kuldeep explained that the export function uses **Canvas rendering**, which can cause performance delays with large data sets.
2. Tom reiterated that backend processing is unnecessary for visual export; it should **simply capture and download what is displayed**.

f. Shift Analytics Segregation (Site, Beat, and Admin)

i. Discussion Points:

1. Jaspreet confirmed that current analytics allow filters for both **Site** and **Beat**-based scheduling.
2. Tom requested that additional segmentation be included:
    a. **All**, **Admin**, **Stationary (previously "Site")**, and **Mobile (previously "Beat")**.
    b. Each category should have its **own section or graph** on the main analytics page.
    c. Optionally, "All" could appear **first** to display combined data.

ii. Clarifications:

1. Jaspreet noted that **Admin shifts** are created separately (for dispatch operations) and not under site-based scheduling.
2. Tom reiterated that all four categories (All, Admin, Stationary, Mobile) should be **visibly present** on the dashboard — this helps when generating **Operations Reports** or taking **printouts** of the full analytics view.

iii. Decisions:

1. Rename and segment categories as:
    a. All
    b. Admin
    c. Stationary (formerly Site)
    d. Mobile (formerly Beat)
2. Each category should display its **own data section** on the dashboard.
3. The "All" section does **not require click-through linking**, as it is for **summary viewing only** (numbers and charts are static).
4. The **pie chart** in this section should be replaced with a **graph**, as Tom preferred bar/line visualization for numerical clarity.

g. Beat Revenue Analytics and Comparison

i. Discussion Points:

1. Gurpreet demonstrated the **Beat Revenue** graph showing:
    a. Current Sites, Previous Sites
    b. Current Hits, Previous Hits
    c. Current Amount, Previous Amount
2. Tom and team found the current view **confusing and mislabeled**.
3. Tom emphasized the report should present **clean, simple numerical summaries** similar to the **reference sheet**

shared by Teresa.

ii. **Feedback Summary:**
1. Randy and Teresa both expressed difficulty understanding the displayed metrics and their relationships.
2. Teresa clarified that the comparison logic should be **month-to-month**, not yearly, to maintain consistent baselines.
3. Tom requested:
    a. Columns for Number of Properties (Sites), Number of Hits, Revenue, and Revenue per Property.
    b. Inclusion of Targets and a Variance (On-Target/Off-Target) column.
    c. Comparison columns should clearly show current vs. previous values with up/down percentages.

iii. **Technical Clarification:**
1. Jaspreet explained the data flow: expanding a beat shows site-level breakdowns with metrics such as **daily hits, rate, and total amount**.
2. Kuldeep confirmed that date selection controls the current and previous period data, but labeling needed improvement for clarity.

iv. **Decisions:**
1. The comparison view will be redesigned for **monthly** comparison only.
2. Each beat should display:
    a. Number of Sites (Properties)
    b. Number of Hits
    c. Revenue
    d. Revenue per Property
    e. Budget/Target and Deficit/Surplus
3. Comparison columns (Previous vs. Current) will show **percentage change**.
4. The view should mimic Teresa's **reference sheet format** shared during the call.
5. Focus on **data accuracy and readability** over visual styling — visuals (graphs) can be added later for summary representation.

h. **Visualization Strategy**

i. **Discussion Points:**
1. Randy recommended that before choosing chart types, the team should first define **what insights** need to be conveyed — then select visualization accordingly.
2. Tom agreed, emphasizing **numeric clarity** over design — "Numbers matter more than looks."
3. Final view should allow decision-making within **seconds**, with key metrics immediately visible.

ii. **Decisions:**
1. Numeric summaries are **mandatory**, graphical elements are **supplementary**.
2. Graphs can be added later for **visual aid**, but not as the primary data source.
3. Focus on **clean data representation** (like the sheet shown by Teresa).