# Document Attributes and Revision History

| Attribute | Value |
| --- | --- |
| Owner | AT&T |
| Additional Description | **AT&T Coding Challenge Summer 2013 Season – Participant Requirements** |
| Other Attribute |  |

## 

## Revision History

| Author | Date | Version # | Revision Description |
| --- | --- | --- | --- |
| AT&T | 6/7/2013 | 1.0 | Baselined requirements – Summer 2013 AT&T Coding Challenge |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**NOTE:** No changes to the document will be allowed when the document is pending approval.

**Table of Contents**

[Document Attributes and Revision History 1](#_Toc302729598)

[Revision History 1](#_Toc302729599)

[1 Overview 4](#_Toc302729600)

[1.1 Application Architecture Overview 4](#_Toc302729601)

[1.2 Network Architecture 5](#_Toc302729602)

[1.3 Technical Specifications 6](#_Toc302729603)

[2 Business Requirements 6](#_Toc302729604)

[3 System Requirements 7](#_Toc302729605)

[3.1 Create Application Logon screen 8](#_Toc302729606)

[3.1.1 Data Elements 8](#_Toc302729607)

[3.1.2 Logic and Validation rule: 8](#_Toc302729608)

[3.2 Create ZIP search screen 9](#_Toc302729609)

[3.2.1 Data Elements 9](#_Toc302729610)

[3.2.2 Logic and Validation rule: 9](#_Toc302729611)

[3.3 Create ZIP search result screen 9](#_Toc302729612)

[3.3.1 Data Elements 9](#_Toc302729613)

[3.3.2 Logic and Validation rule: 10](#_Toc302729614)

[3.4 Create GLC search screen 10](#_Toc302729615)

[3.4.1 Data Elements 11](#_Toc302729616)

[3.4.2 Logic and Validation rule: 11](#_Toc302729617)

[3.5 Create GLC search result screen with “OPEN” Transaction 11](#_Toc302729618)

[3.5.1 Data Elements 11](#_Toc302729619)

[3.3.2 Logic and Validation rule: 11](#_Toc302729620)

[3.6 Create GLC search result screen with “NO” Transaction 12](#_Toc302729621)

[3.6.1 Data Elements 12](#_Toc302729622)

[3.4.2 Logic and Validation rule: 12](#_Toc302729623)

[3.7 Create new damage assessment screen 12](#_Toc302729624)

[3.7.1 Data Elements 12](#_Toc302729625)

[3.5.2 Logic and Validation rule: 16](#_Toc302729626)

[3.8 Create Update damage assessment screen 16](#_Toc302729627)

[3.8.1 Data Elements 16](#_Toc302729628)

[3.6.2 Logic and Validation rule: 20](#_Toc302729629)

[3.9 Create Reports selection screen 20](#_Toc302729630)

[3.9.1 Data Elements 20](#_Toc302729631)

[3.7.2 Logic and Validation rule: 20](#_Toc302729632)

[3.10 Create Admin Condition Canned Report 21](#_Toc302729633)

[3.10.1 Data Elements 21](#_Toc302729634)

[3.8.2 Logic and Validation rule: 21](#_Toc302729635)

[3.11 Create CRE Building Closure/Delayed Open Canned Report 21](#_Toc302729636)

[3.11.1 Data Elements 21](#_Toc302729637)

[3.9.2 Logic and Validation rule: 22](#_Toc302729638)

[4 Backend APIs 22](#_Toc302729639)

[4.1 Application Logon 22](#_Toc302729640)

[4.2 List Incidents By GEOLOC 24](#_Toc302729641)

[4.3 List Buildings by Zip 25](#_Toc302729642)

[4.4 Retrieve Incident by Record Number 27](#_Toc302729643)

[4.5 Update Incident by Record Number 28](#_Toc302729644)

[4.6 Add Incident by GEOLOC 29](#_Toc302729645)

[4.7 Get Admin Condition Report Data 30](#_Toc302729646)

[4.8 Get CRE Building Closure/Delayed Opening Data 32](#_Toc302729647)

[4.9 Add Incident Picture by Record Number 34](#_Toc302729648)

[4.10 Delete Incident Picture by FileName and Record Number 35](#_Toc302729649)

[4.11 Get Incident Picture by Record Number 36](#_Toc302729650)

[4.12 Incident Structure Definition 36](#_Toc302729651)

[4.13 Status Code definitions 43](#_Toc302729652)

[4.14 Reference Data 44](#_Toc302729653)

[4.15 Sample SOAP requests/responses 44](#_Toc302729654)

[5 Appendix 63](#_Toc302729655)

[5.1 Glossary 63](#_Toc302729656)

# Overview

The purpose of the AT&T Coding Challenge is to establish a software development competition among universities. The selected solution may be used within AT&T’s operating environment.

This document specifies the requirements for the mobile application to be developed during the challenge.

## Application Architecture Overview

The use case for season #1 calls for the development of mobile client software for the Disaster Reporting Tool (DRT) used by the Corporate Real Estate (CRE) Property Management organization. The software will enable CRE Property Managers to collect damage information via a mobile device.

**Background**

CRE’s Disaster Reporting Tool System (DRT) does not support mobile users. The students will build new software for a mobile device (i.e. Android phone) that will enable the Property Managers to use a mobile version of this system.

**Application Architecture**

Figure 1 depicts the high level software architecture and development scope.

# 

Figure 1 - DRT Wireless client

The students will be responsible for developing the software that will run inside the mobile device (e.g. mobile application) as shown on Figure 1. AT&T will provide a backend environment that will expose the necessary transactions (i.e. Web Services) to support the mobile client.

## Network Architecture

The AT&T Coding Challenge will leverage the following infrastructure components as its operating environment.

* **AT&T Firewall** – AT&T CSO reverse proxy used to provide secured access to devices outside of the firewall which need access to the AT&T backend.
* **AT&T Coding Challenge Backend server** –hosts the backend software supporting the mobile device. This is already configured.

Figure 2 shows the network access path that the mobile devices will follow as part of the competition and the dry-run.

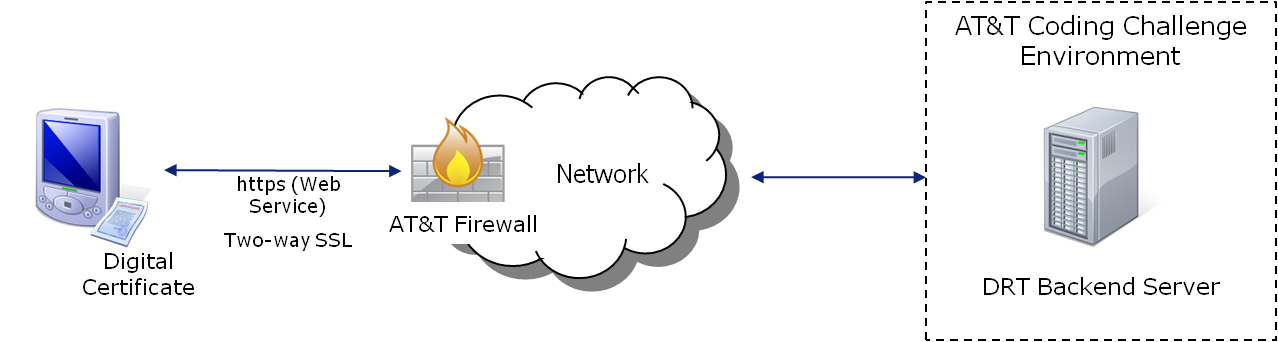


Figure 2 - Device Access Path

**Access flow**

1. Mobile device makes an HTTPS request to invoke a Web service (i.e. SOAP over HTTPS) or an HTTPS GET to retrieve a picture (see backend requirements for more details). The device will use a particular DNS hostname/port to access the AT&T Coding Challenge environment.
2. The AT&T firewall receives the HTTPS request from the Mobile client and performs a two-way SSL authentication (i.e. client certificate authentication).
3. If Access is allowed, then AT&T Firewall will forward the request to AT&T Coding challenge environment.
4. The Mobile client shall then receive the reply from AT&T Coding Challenge server.

## Technical Specifications

The mobile client shall have access to various AT&T APIs to implement the business requirements to be stated as part of the competition. Figure 3 depicts the architectural components of system in more detail.

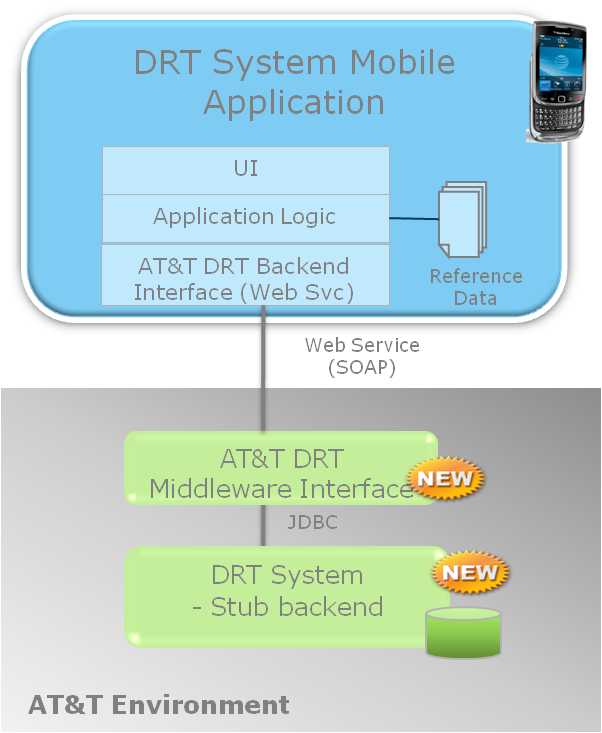


Figure 3 – CRE-DRT Mobile Client Architecture

# Business Requirements

| Unique ID | Business Requirement | Status  (Proposed, Defective, Rejected, Deferred or Approved) |
| --- | --- | --- |
| BR-1.A | New Mobile Assessment of Storm Damaged Property  Provide the ability for the mobile users to log into DRT application | Approved |
| BR-1.B | Display an initial incident search screen  Users shall search incidents by providing the GEOLOC of a property or a ZIPCODE. | Approved |
| BR-1.C | Display the results of the incident when searching by GEOLOC  The application shall determine if any incidents have been submitted and shall display them. If there are no prior reports submitted, provide the ability to create a new report. If prior report submission exists, provide the ability to select it. | Approved |
| BR-1.D | Display the results of the building search by ZIPCODE  The application shall provide a list of buildings within a ZIPCODE area. The information of each building shall include a GEOLOC, which can be use to search for incidents related to that location (i.e. BR-1.C). | Approved |
| BR-1.E | Validate Property (GEOLOC) provided by the user.  Each Property-related information (GEOLOC) provided by the user shall represent a valid AT&T location. | Approved |
| BR-1.F | Provide the ability to add a Damage Assessment Record (i.e. incident)  The user shall be able to provide information about the incident including pictures taken from the device. | Approved |
| BR-2.A | Provide the ability to update an existing Damaged Assessment Record (i.e. incident) | Approved |
| BR-3.A | Provide the ability to run Reports  The mobile application shall provide the means of running the following reports: 1) Admin Condition Report and 2) CRE Building Closure/Delayed Open Report. | Approved |

# System Requirements

Utilize Android SDK and customized backend API to replicate the functionalities of AT&T DRT application on mobile environment. Mobile application should be able to allow authenticate user access right, update/create assessments and generate reports.

| Unique ID | System Requirement | Business Requirement | Status  (Proposed, Defective, Rejected, Deferred or Approved) |
| --- | --- | --- | --- |
| SR-1.A | Create Application Logon Screen (See section 3.1 for details) | BR-1.A | Approved |
| SR-1.B | Create GEOLOC Search Screen (See section 3.3 for details) | BR-1.B | Approved |
| SR-1.C | Create GEOLOC Search Result with OPEN Transaction (See section 3.4 for details) | BR-1.D  BR-1.E | Approved |
| SR-1.D | Create GEOLOC Search Result NO Transaction (See section 3.5 for details) | BR-1.D  BR-1.E | Approved |
| SR-1.E | Create a new damage assessment screen (See section 3.6 for details) | BR-1.F | Approved |
| SR-1.F | Create ZIPCODE Search Screen (See section 3.2 for details) | BR-1.B  BR-1.C | Approved |
| SR-2.A | Create a Update damage assessment screen (See section 3.7 for details) | BR-2.A | Approved |
| SR-3.A | Create a Reports selection screen (See section 3.8 for details) | BR-3.A | Approved |
| SR-3.B | Create an Admin Condition Canned Report screen (See section 3.9 for details) | BR-3.A | Approved |
| SR-3.C | Create a CRE Building Closure/Delayed Open Canned Report screen (See section 3.10 and 3.11 for details) | BR-3.A | Approved |

## Create Application Logon screen

* Allow user to enter User ID and Password to authenticate user access right
* Retrieve “Authorization Type”, “Session Token ID” and “Timestamp”
* Application timeout at 15 minutes with no activities from token and return to Logon screen

### Data Elements

| Data Elements | Input Object Type/Size | Comments |
| --- | --- | --- |
| UserID | Must be 6 characters |  |
| Password | Must be between 6 to 12 characters |  |

### Logic and Validation rule:

If UserID does not exist

Display Message and exit

Else

Get Authorization type from “Application Logon“ API

Get session Token ID

Store session Token ID and Timestamp

Store UserID and Authorization type to session variable

If Authorization type = “ADM”

Create link launch GLC search screen

Create link to launch canned report filter screen

Else if Authorization type = “RPT”

Launch canned report filter screen

## Create ZIP search screen

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Enter ZIPCODE to search”)
* Retrieve “Authorization Type”, “Session Token ID” and “Timestamp”
* Application timeout at 15 minutes with no activities from token and return to Logon screen
* Display “UserID” and “Authorization type” on UI. It must be disabled

### Data Elements

| Data Elements | Input Object Type/Size | Comments |
| --- | --- | --- |
| ZIP | Numeric, 5 digits | Input field to retrieve a list of buildings |

### Logic and Validation rule:

If ZIP Input field == BLANK

Display error message and return to ZIP search screen

Else

Pass ZIP to “List Buildings by ZIP” API to get transaction information

If Data found THEN

Display Zip search result screen

END IF

END IF

## Create ZIP search result screen

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Search result”)
* Display all data element with header column
* Provide the user with the ability to search for incidents by GEOLOC based on any of the buildings provided.

### Data Elements

| Data Elements | Comments |
| --- | --- |
| GLC | Display only |
| Building | Display only |
| State | Display only |
| PM | Display only |
| Longitude | Not displayed |
| Latitude | Not displayed |
| GET INCIDENT(S) | Button (or similar UI element) that performs a GEOLOC incident search |

### Logic and Validation rule:

If selected ZIPCODE has buildings associated

Populate grid with required building information

Else

Display error message and return to ZIPCODE search screen

END-IF

## Create GLC search screen

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Enter GLC to search”) to present the UI or area users is working on
* Retrieve “Authorization Type”, “Session Token ID” and “Timestamp”
* Application timeout at 15 minutes with no activities from token and return to Logon screen
* Display “UserID” and “Authorization type” on UI. It must be disabled

### Data Elements

| Data Elements | Input Object Type/Size | Comments |
| --- | --- | --- |
| GLC | Must be between 4 to 12 characters | Input field to retrieve Building information |

### Logic and Validation rule:

If GLC Input field == BLANK

Display error message and return to GLC search screen

Else

Pass GLC to “List Incidents by GEOLOC” API to get transaction information

If selected GLC has transaction and status is “OPEN”

Launch “UPDATE” Damage Assessment screen

Else

Launch “ADD” Damage Assessment screen

## Create GLC search result screen with “OPEN” Transaction

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Search result”) to present the UI or area users is working on
* Display all data element with header column
* Create update event for “Update” on data element “Action”

### Data Elements

| Data Elements | Comments |
| --- | --- |
| Action | Action Type and Event driven by transaction data |
| Rec # | Display only |
| GLC | Display only |
| Building | Display only |
| Hurricane | Display only |
| Status | Display only |
| WR | Display only |

### 3.5.2 Logic and Validation rule:

If selected GLC has transaction and status is “OPEN”

Populate grid with transaction

Populate column “Action” with “Update” event button

If “Update” button clicked

Pass record number to “Retrieve Incident by Record Number” API to get transaction information

Launch update damage assessment screen

Else

Do nothing screen

## Create GLC search result screen with “NO” Transaction

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Search result”) to present the UI or area users is working on
* Display all data element with header column
* Create Add event for “Add” on data element “Action”

### Data Elements

| Data Elements | Comments |
| --- | --- |
| Action | Action Type and Event driven by transaction data |
| GLC | Display only |
| Building | Display only |
| State | Display only |
| PM | Display only |

### Logic and Validation rule:

If selected GLC exist

Populate column “Action” with “ADD” event button

Populate grid with required building information

Else

Display error message and return to GLC search screen

If “ADD” button clicked

Store all building information into session variable to use in UI field

Launch new damage assessment screen

Else

Do nothing

## Create new damage assessment screen

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Search result”) to present the UI or area users is working on
* Display all data element with Label
* Create “GET” and “ADD” event and button

### Data Elements

| **Label** | **Input Object Type/Size** | **Default Value** | **Editable** | **Comments** |
| --- | --- | --- | --- | --- |
| GLC: | Text  [VARCHAR2(15 BYTE)] | Selected GLC | N | * object disabled * Value from GLC search screen entry |
| State: | Text  [VARCHAR2(2 BYTE)] | Selected GLC's State | N | * object disabled * Get from building information session variable |
| PM: | Text  [VARCHAR2(50 BYTE)] | Selected GLC's PM | N | * object disabled * Get from building information session variable |
| Building Name: | Text  [VARCHAR2(50 BYTE)] | Selected GLC's Building Name | N | * object disabled * Get from building information session variable |
| Address: | Text  [VARCHAR2(255 BYTE)] | Selected GLC's Address | N | * object disabled * Get from building information session variable |
| Event Year: | Drop Down List Box  [VARCHAR2(4 BYTE)] | Current Year | Y | Require field |
| Requestor UserID: | Text  [VARCHAR2(6 BYTE)] | Current Login UserID | Y | UserID is from stored session variable |
| Contact #: | Text  [VARCHAR2(10 BYTE)] | Blank | Y | * Must include phone format (###)-###-#### * require field |
| Initial Report Date: | Text  [DATE] | Current Date | Y | * Current Date and in date format MM/DD/YYY * Required field |
| Event: | Drop Down List Box  [VARCHAR2(30 BYTE)] | "Select Storm Type of Hurricane Name" | Y | * Event is driven by Year * Hardcoded into Drop Down Lists from reference file |
| Property Type: | Drop Down List Box  [VARCHAR2(10 BYTE)] | "ADM" | Y | * Manual coded into Drop Down Lists from reference file |
| Work Request #: | Text  [VARCHAR2(50 BYTE)] | Blank | Y |  |
| Non-CO Facilities ONLY: |  |  |  | Group Header |
| Commercial Power: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Y | Under "Non-CO Facilities ONLY:" group |
| On Generator: | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Under "Non-CO Facilities ONLY:" group |
| Any Facility: |  |  |  | Group Header |
| Damaged: | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Under "Any Facility:" group |
| Mobility CO: | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Under "Any Facility:" group |
| Unoccupiable: | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Under "Any Facility:" group |
| Building Status: | Drop Down List Box  [VARCHAR2(20 BYTE)] | OPEN | Y | Option available for DropDownListBox are “OPEN”, “CLOSED”, “DELAYED OPENING” and “RE-OPENED” |
| CRE Lead: | Drop Down List Box  [VARCHAR2(10 BYTE)] | PM | Y | Option available for DropDownListBox are “PM” and “D&C” |
| Estimated Costs: |  |  |  | Group Header |
| Capital ($): | Text  [NUMBER(16,2)] | 0 | Y | Under "Estimated Costs:" group |
| Expense ($): | Text  [NUMBER(16,2)] | 0 | Y | Under "Estimated Costs:" group |
| Categories: |  |  |  | Group Header |
| Electrial: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Electrical:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Electrical:" |
| Environmental: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Environmental:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Environmental:" |
| Fence/Gates: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Fence/Gates:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Fence/Gates:" |
| Generator: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Generator:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Generator:" |
| Water: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Water:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Water:" |
| Grounds: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Grounds:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Grounds:" |
| Mechanical: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Mechanical:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Mechanical:" |
| Other: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Other:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Other:" |
| Progress: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Progress:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Progress:" |
| Plumbing: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Plumbing:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Plumbing:" |
| Roofs: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Roofs:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Roofs:" |
| Safety: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Safety:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Safety:" |
| Structural: |  |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Structural:" |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | N | Y | Subgroup of "Structural:" |
| Assessment: | Text  [VARCHAR2(255 BYTE)] | Blank | Y |  |
| Status: | Text  [VARCHAR2(255 BYTE)] | Blank | Y |  |
| Report Open/Closed: | Drop Down List Box  [VARCHAR2(6 BYTE)] | Open | Y |  |
| Completion Date: | Text  [DATE] | Current Date | Y | * Default to Initial Date + 1 or Manual enter Date * Date format MM/DD/YYYY * require field |
| Notes: | Text  [VARCHAR2(255 BYTE)] | Blank | Y |  |

### Logic and Validation rule:

Disable GET picture event button

Enable ADD picture event button

If “ADD” event click

Pass Record Number to “Add Incident Picture by Record Number” API (if the incident was created)

If the pictures counter for current Work Request # <= 10

Enable GET button

Capture picture and use “Add Incident Picture by Record Number” API to store picture

Else

Display message “Pictures limit reached”

If “GET” event click

Launch a picture screen and use “Get Incident Picture by Record Number” API to view individual pictures

If transaction status is OPEN

Allow user to use “Delete Incident Picture by FileName and Record Number” API to DELETE pictures

## Create Update damage assessment screen

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Search result”) to present the UI or area users is working on
* Display all data element with label
* Create “GET” and “ADD” event and button

### Data Elements

| **Label** | **Input Object Type** | **Editable** | **Comments** |
| --- | --- | --- | --- |
| Record ID: | Text  [VARCHAR2(25 BYTE)] | N | object disabled |
| CRE Lead: | Text  [VARCHAR2(3 BYTE)] | N | object disabled |
| Event: | Text  [VARCHAR2(50 BYTE)] | N | object disabled |
| Event Year: | Text  [VARCHAR2(4 BYTE)] | N | object disabled |
| State: | Text  [VARCHAR2(2 BYTE)] | N | object disabled |
| PM: | Text  [VARCHAR2(50 BYTE)] | N | object disabled |
| GLC: | Text  [VARCHAR2(15 BYTE)] | N | object disabled |
| Property Type: | Text  [VARCHAR2(10 BYTE)] | N | object disabled |
| Building: | Text  [VARCHAR2(50 BYTE)] | N | object disabled |
| Address: | Text  [VARCHAR2(255 BYTE)] | N | object disabled |
| Requestor: | Text  [VARCHAR2(6 BYTE)] | N | object disabled |
| Contact #: | Text  [VARCHAR2(10 BYTE)] | N | object disabled |
| Initial Report Date: | Text  [DATE] | N | object disabled |
| Progress: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Electrical: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Grounds: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Plumbing: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Environmental: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Mechanical: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Roofs: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Fence/gates: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Other: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Safety: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Generator: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Structural: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Water: | Text  [VARCHAR2(1 BYTE)] | N | object disabled |
| Work Request #: | Text  [VARCHAR2(50 BYTE)] | N | object disabled |
| Assessment: | Text  [VARCHAR2(255 BYTE)] | Y |  |
| Status: | Text  [VARCHAR2(255 BYTE)] | Y |  |
| Completion Date: | Text  [DATE] | Y | Require field |
| Report Open/Closed: | Drop Down List Box  [VARCHAR2(6 BYTE)] | Y |  |
| Notes: | Text  [VARCHAR2(255 BYTE)] | Y |  |
| Non-CO Facilities ONLY: |  |  | Group Header |
| Commercial Power: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Under "Non-CO Facilities ONLY:" group |
| On Generator: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Under "Non-CO Facilities ONLY:" group |
| Any Facility: |  |  | Group Header |
| Damaged: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Under "Any Facility:" group |
| Mobility CO: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Under "Any Facility:" group |
| Not occupiable: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Under "Any Facility:" group |
| Building Status: | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y |  |
| Estimated Costs: |  |  | Group Header |
| Capital ($): | Text  [NUMBER(16, 2)] | Y | Under "Estimated Costs:" group |
| Expense ($): | Text  [NUMBER(16, 2)] | Y | Under "Estimated Costs:" group |
| Categories: |  |  | Group Header |
| Electrial: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Electrical:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Electrical:" |
| Environmental: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Environmental:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Environmental:" |
| Fence/Gates: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Fence/Gates:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Fence/Gates:" |
| Generator: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Generator:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Generator:" |
| Water: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Water:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Water:" |
| Grounds: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Grounds:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Grounds:" |
| Mechanical: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of " Mechanical:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Mechanical:" |
| Other: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Other:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Other:" |
| Plumbing: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Plumbing:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Plumbing:" |
| Roofs: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Roofs:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Roofs:" |
| Safety: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Safety:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Safety:" |
| Structural: |  |  |  |
| Issue | Drop Down List Box  [VARCHAR2(1 BYTE)] | C | * Subgroup of "Structural:" * If Value = “N” then object enable and editable else disable |
| Closed | Drop Down List Box  [VARCHAR2(1 BYTE)] | Y | Subgroup of "Structural:" |

### Logic and Validation rule:

Enable GET and ADD picture event button

If “ADD” event click

Pass Record Number to “Add Incident Picture by Record Number” API to create folder

If the pictures counter for current Work Request # <= 10

Enable ADD Picture button

Capture picture and use “Add Incident Picture by Record Number” API to store picture

Else

Display message “Pictures limit reached”

If “GET” event click

Launch a picture screen and use “Get Incident Picture by Record Number” API to view individual pictures

If transaction status is OPEN

Allow user to use “Delete Incident Picture by FileName and Record Number” API to DELETE pictures

## Create Reports selection screen

* Screen must contain corresponded Header and Sub header (ex: header will be “Damage Assessment Reporting” and Sub header will be ”Search result”) to present the UI or area users is working on
* Display all data element with label
* Link to open “Admin Condition Canned Report”
* Link to open “CRE Building Closure/Delayed Open Canned Report”

### Data Elements

|  |  |  |  |
| --- | --- | --- | --- |
| **Label** | **Input Object Type** | **Default Value** | **Comments** |
| Event Year: | Drop Down List Box  [VARCHAR2(4 BYTE)] | Current Year | Require field |
| Event Name: | Drop down List Box  [VARCHAR2(30 BYTE)] | Y | * Event is driven by Year. |

### Data Elements

Create Event Year Drop Down List Box and Use reference file to hardcoded event year into Drop Down Lists

Create Event Name Drop Down List Box and use reference file to hardcoded event name into Drop Down Lists

## Create Admin Condition Canned Report

* Screen must contain corresponded Header and Sub header (ex: header will be “Admin Condition Canned Report” and Sub header will be ”Damage Report results”) to present the UI or area users is working on
* Display all data elements with label

### Data Elements

|  |  |
| --- | --- |
| **Label** | **Output Data field** |
| Event: | [VARCHAR2(30 BYTE)] |
| Event Year: | [VARCHAR2(4 BYTE)] |
| GLC: | [VARCHAR2(15 BYTE)] |
| Building: | [VARCHAR2(50 BYTE)] |
| Address: | [VARCHAR2(225 BYTE)] |
| STATE: | [VARCHAR2(2 BYTE)] |
| Type: | [VARCHAR2(10 BYTE)] |
| PM: | [VARCHAR2(50 BYTE)] |
| Assessment: | [VARCHAR2(255 BYTE)] |
| Status: | [VARCHAR2(255 BYTE)] |
| Damaged: | [VARCHAR2(1 BYTE)] |
| Commerical Power: | [VARCHAR2(1 BYTE)] |
| On Generator: | [VARCHAR2(1 BYTE)] |
| Un-Occupiable: | [VARCHAR2(1 BYTE)] |

### Logic and Validation rule:

Pass event Year and event name from filter criteria screen to “Get Admin Condition Report Data” API to get Admin Condition result set

## Create CRE Building Closure/Delayed Open Canned Report

* Screen must contain corresponded Header and Sub header (ex: header will be “CRE Building Closure/Delayed Open Canned Report” and Sub header will be ”Damage Report Results”) to present the UI or area users is working on
* Display all data elements with label

### Data Elements

|  |  |
| --- | --- |
| **Label** | **Output Data field/Data Type** |
| Event: | [VARCHAR2(30 BYTE)] |
| Initial Request Date: | [DATE] |
| Building Status: | [VARCHAR2(20 BYTE)] |
| GLC: | [VARCHAR2(15 BYTE)] |
| Address: | [VARCHAR2(255 BYTE)] |
| STATE: | [VARCHAR2(2 BYTE)] |
| Type: | [VARCHAR2(10 BYTE)] |
| PM: | [VARCHAR2(50 BYTE)] |
| Assessment: | [VARCHAR2(255 BYTE)] |
| Status: | [VARCHAR2(255 BYTE)] |

### Logic and Validation rule:

Pass event Year and event name from filter criteria screen to “Get CRE Building Closure/Delayed Opening data” API to get CRE Building Closure/Delayed Open result set

# Backend APIs

The mobile client and the middleware layer will use a synchronous request/response model. The API will be exposed in the form of SOAP Web Services. The following is a list of the transactions available to the client.

1. Application Logon
2. List Incidents by GEOLOC
3. List Incidents by Zip
4. Retrieve Incident by Record Number
5. Update Incident by Record Number
6. Add Incident by GEOLOC
7. Get Admin Condition Report Data
8. Get CRE Building Closure/Delayed Opening Data
9. Add Incident Picture by Record Number
10. Delete Incident Picture by Picture ID
11. Get Incident Picture by Record Number

**Flow constraints**

Transaction #1 (Application Logon) must be invoked before any other transaction in order to authenticate, authorize and establish a session token for the mobile user. The session token must be included on all other transactions. This token will be used to keep track of the mobile user activity. If there is no activity within the timeout period (i.e. 15 minutes), then the AT&T middleware will reject all further transitions. The mobile shall then re-authenticate the user.

## Application Logon

| Operation | | Description | |
| --- | --- | --- | --- |
| Application Logon | | * Allow the user to authenticate with the backend | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| User Name | String (6) | User ID of the mobile user | WSDL name: username |
| Password | String (12) | User password | Suggestion:  MD5 Hash of the password  WSDL name:  password |
| **Output** | | | |
| Token | String(25) | Security token generated by the system (random number sequence).  Note that the token should expire after 15 mins of inactivity. | To be provided on all other transactions  WSDL name: token |
| Authorization | String(4) | Depicts the level of user authorization. | * ADM – user is fully authorized to use the system. * RPT – user can only use the report capabilities. * WSDL name: authorization |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## List Incidents By GEOLOC

| Operation | | Description | |
| --- | --- | --- | --- |
| List Incidents by GEOLOC | | * Retrieve a list of incidents based on a GEOLOC as input. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(25) | Security token provided by the application Logon service | WSDL name: token |
| GEOLOC | String(15) | GEOLOC field | WSDL name: geoLoc |
| **Output** | | | |
| List of Incidents [1 – n]. Includes the following fields:  If incidents are found the following fields will be populated:   * RecNumber * GEOLOC * Building Name * Building Address * Event Name * Incident Status * Work Request Number * State * PM   Otherwise location information will be provided if the GEOLOC is valid   * GEOLOC * Building Name * State * PM | | | |
| RecNumber | Number(11) | Identifier associated with each incident. | * WSDL name: incidentList>recNumber |
| Building Name | String(50) | Name of the building | * WSDL name: incidentList>buildingName |
| Building Address | String(255) | Address of the building | Screen name: Address  WSDL name: incidentList>buildingAddress |
| Event Name | String(50) | Identifies the type of incident. | Screen name: Event  WSDL name: incidentList>eventName |
| Incident Status | String(10) | Identifies the status of the incident. | * OPEN * CLOSED   WSDL name: incidentList>incidentStatus |
| Work Request Number | Number (50) | Work Request number associated with the incident | WSDL name: incidentList>workReqNumber |
| State | String(2) | State associated with the location | WSDL name: incidentList>state |
| Property Manager ATTUID | String(50) | AT&T User ID of the Property Manager. | WSDL name: incidentList>pmATTUID |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## List Buildings by Zip

| Operation | | Description | |
| --- | --- | --- | --- |
| List Buildings by ZIP | | * Retrieve a list of buildings based on a ZIP CODE as input. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(25) | Security token provided by the applicationLogon service | WSDL name: token |
| ZIPCODE | Number(5) | ZIPCODE field | WSDL name: Zipcode |
| **Output** | | | |
| List of Buildings [1 – n]. Includes the following fields:   * GEOLOC * Building Name * Building Address * State * PM * Longitude * Latitude | | | |
| GEOLOC | String(15) | GEOLOC field | WSDL name: geoLoc |
| Building Name | String(50) | Name of the building | WSDL name: incidentList>buildingName |
| Building Address | String(255) | Address of the building | Screen name: Address  WSDL name: incidentList>buildingName |
| State | String(2) | State associated with the location | WSDL name: incidentList>state |
| Property Manager ATTUID | String(50) | AT&T User ID of the Property Manager. | WSDL name: incidentList>pmATTUID |
| Longitude | Number | Longitude of the building | This field is not available for all buildings.  WSDL name: incidentList>Longitude |
| Latitude | Number | Latitude of the building | This field is not available for all buildings.  WSDL name: incidentList>Latittude |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## Retrieve Incident by Record Number

| Operation | | Description | |
| --- | --- | --- | --- |
| Retrieve Incident by Record Number | | * Retrieve a specific incident using a Record Number as input. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(25) | Security token provided by the applicationLogon service | WSDL name: token |
| RecNumber | Number(11) | Identifier associated with each incident. | WSDL name: reqNumber |
| **Output** | | | |
| Incident [1] (One instance of the Incident Structure. See section 4.124.12 definition of the Incident structure for more information)  WSDL name: incidentStructure | | | |
| List of Image URIs [0 – 10]. Includes the following field:   * Image URI   WSDL name: imageURI | | | |
| Image URI | String(100) | URI to the path where the image is stored on the web server | Note that this information is not stored on the DB and should be constructed based on images available.  WSDL name: imageURI |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## Update Incident by Record Number

| Operation | | Description | |
| --- | --- | --- | --- |
| Update Incident by Record Number | | * Updates a specific incident using a Record Number as input. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(25) | Security token provided by the application Logon service | WSDL name: token |
| RecNumber | Number(11) | Identifier associated with each incident. | WSDL name: recNumber |
| Incident [1] (One instance of the Incident Structure. See section 4.124.12 definition of the Incident structure for more information)  WSDL name: incidentStructure | | | |
| **Output** | | | |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## Add Incident by GEOLOC

| Operation | | Description | |
| --- | --- | --- | --- |
| Adds Incident by GEOLOC | | * Adds an incident using a GEOLOC as input. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(25) | Security token provided by the applicationLogon service | WSDL name: token |
| Incident [1] (One instance of the Incident Structure. See section 4.124.12 definition of the Incident structure for more information)  WSDL name: incidentStructure | | | |
| **Output** | | | |
| RecNumber | Number(11) | Identifier associated with each incident. | WSDL name: recNumber |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## Get Admin Condition Report Data

| Operation | | Description | |
| --- | --- | --- | --- |
| Get Admin Condition Report Data | | * Gets information about the condition of admin buildings that are affected based on a year and/or type of incident. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(20) | Security token provided by the applicationLogon service | WSDL name: token |
| Year | String(4) |  | WSDL name: year |
| Event Name | String(50) | Identifies the type of incident. | Screen name: Event  WSDL name: eventName |
| **Output** | | | |
| RecNumber | Number(11) | Identifier associated with each incident. | WSDL name: recNumber |
| GEOLOC | String(15) | GEOLOC field related to the incident. | WSDL name: geoLoc |
| Incident Year | String(4) | Year that the incident took place. | Format:  yyyy  Screen name: Event Year  WSDL name: incidentYear |
| Building Name | String(50) | Name of the building | WSDL name: buildingName |
| Building Address | String(255) | Address of the building | Screen name: Address  WSDL name: buildingAddress |
| Event Name | String(50) | Identifies the type of incident. | Screen name: Event  WSDL name: eventName |
| State | String(2) | State associated with the location | WSDL name: state |
| Property Manager ATTUID | String(50) | AT&T User ID of the Property Manager. | WSDL name: pmATTUID |
| Assessment Notes | String (1000) | Notes providing an assessment of the incident. | WSDL name: assessNotes |
| Status Notes | String (1000) | Notes providing status of the incident. | WSDL name: statusNotes |
| Damage Indicator | Char(1) | Indicates whether there as any damage to the property or not. | Values:   * Y * N   WSDL name: damageIndicator |
| Commercial Power Indicator | Char(1) | Indicates whether there is commercial power or not. | Values:   * Y * N   WSDL name: comPowerIndicator |
| On Generator Indicator | Char(1) | Indicates whether there the property is on a generator or not. | Values:   * Y * N   WSLD name: onGeneratorIndicator |
| Un-Occupiable Indicator | Char(1) | Indicates whether there the property is occupiable or not. | Values:   * Y * N   WSDL name: unOccupiableIndicator |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMessage |

## Get CRE Building Closure/Delayed Opening Data

| Operation | | Description | |
| --- | --- | --- | --- |
| Get CRE Building Closure/Delayed Opening data. | | * Gets information about the Building that are closed or have a delayed opening | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(20) | Security token provided by the applicationLogon service | WSDL name: token |
| Year | String(4) |  | WSDL name: year |
| Event Name | String(50) | Identifies the type of incident. | Screen name: Event  WSDL name: eventName |
| **Output** | | | |
| RecNumber | Number(11) | Identifier associated with each incident. | WSDL name: recNumber |
| GEOLOC | String(15) | GEOLOC field related to the incident. | WSDL name: geoLoc |
| Event Name | String(50) | Identifies the type of incident. | * HURRICANE * TORNADO * FLOOD * WILDFIRE   Screen name: Event  WSDL name: eventName |
| Initial Report Date | Date | Date when the incident was initially reported | Format:  mm/dd/yyyy  WSDL name: initialRptDate |
| Building Status | Char(20) | Indicated the availability of the facility. | Values:   * OPENED * CLOSED * DELAYED OPENING * RE-OPENED   WSDL name: buildingStatus |
| Building Type | String(10) | Indicates the type of building | Values:   * ADM * CO   Screen name: Type  WSDL name: buildingType |
| Building Address | String(255) | Address of the building | WSDL name: buildingAddress |
| State | String(2) | State associated with the location | WSDL name: state |
| Property Manager ATTUID | String(6) | AT&T User ID of the Property Manager. | WSDL name: PMAttuid |
| Assessment Notes | String (1000) | Notes providing an assessment of the incident. | WSDL name:  Assessnotes |
| Status Notes | String (1000) | Notes providing status of the incident. | WSDL name: statusNotes |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  WSDL name: requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  WSDL name: requestStatus>statusMesssage |

## Add Incident Picture by Record Number (Bonus)

| Operation | | Description | |
| --- | --- | --- | --- |
| Add Incident Picture by Record Number | | * Stores a picture associated to an incident. * Note that this operation follows the “SOAP with attachment” specification to include the image file needed by this transaction.   <http://www.w3.org/TR/SOAP-attachments#SOAPReferenceToAttachements>   * Per the specification, the operation expects a multi-part message. The first message is SOAP payload. The second message is the actual image file. * See section 4.14 for an example. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| **Fields of SOAP payload (part #1)** | | | |
| Token | String(20) | Security token provided by the applicationLogon service | WSDL name: token |
| RecNumber | Number(11) | Identifier associated with each incident. | WSDL name: recNumber |
| FileName | String(50) | Name of the JPEG file. | WSDL name: fileName |
| Picture | Reference to Binary attachment | Reference to JPEG file associated with the RecNumber. | Note that the actual image file must be included as part of the second part of the message.  WSDL name: picture |
| **Image payload (part #2)** | | | |
| Raw Image file | | | |
| **Output** | | | |
| Image URI | String(100) | URI to the path where the image is stored on the web server | WSDL name: imageURI |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  requestStatus>statusMesssage |

## Delete Incident Picture by FileName and Record Number (Bonus)

| Operation | | Description | |
| --- | --- | --- | --- |
| Delete Incident Picture by FileName and Record Number | | * Deletes a specific picture related to an incident. | |
| **Input** | | | |
| Field Name | Field Type | Description | Notes/Values |
| Token | String(20) | Security token provided by the applicationLogon service | WSDL name: token |
| FileName | String(100) | Filename of the image file. | WSDL name: fileName |
| RecNumber | Number(11) | Record ID of the incident | WSDL name: recNumber |
| **Output** | | | |
| Request Status [1]. Includes the following fields:   * Status Code * Status Message   WSDL name: requestStatus | | | |
| Status Code | Number | Code depicting the success of failure of the transaction. | See section 4.12 for more information.  requestStatus>statusCode |
| Status Message | String(100) | Additional field providing a narrative | See section 4.12 for more information.  requestStatus>statusMesssage |

## Get Incident Picture by Record Number (Bonus)

There will be no specific web service operation required to retrieve the pictures. The pictures will be strategically placed on directory accessible via the web server. The mobile client will issue an HTTP GET request to retrieve the pictures based on the URI provided by the Retrieve incident transaction.

Format

<https://hostname:port/incident_number/image_filename.jpg>

e.g.

https://attcodingchallenge.att.com:82/101010/val.jpg

## Incident Structure Definition

| **Incident Structure** | | | |
| --- | --- | --- | --- |
| Field Name | Field Type | Description | Notes/Values |
| GEOLOC | String(15) | GEOLOC field | WSDL name: geoLoc |
| Event Name | String(50) | Identifies the type of incident. | Screen name: Event  WSDL name: eventName |
| Incident Status | Char(10) | Identifies the status of the incident. | * OPEN * CLOSED   Screen name: Report Open/closed  WSDL name: incidentStatus |
| Incident Completion Date | Date | Date that the incident report was completed. | Screen name: Completion Date  WSDL name: incidentCompltnDate |
| Work Request Number | Number (50) | Work Request number associated with the incident | WSDL name: workReqnumber |
| Incident Year | String(4) | Year that the incident took place. | Format:  yyyy  Screen name: Event Year  WSDL name: incidentYear |
| Requestor ATTUID | String(50) | AT&T User ID of the requestor | WSDL name: reqATTUID |
| Contact Phone | String(10) | Telephone Number of a contact person associated with the incident. | Telephone number representation with no parenthesis or dashes.  WSDL name: contactPhone |
| Initial Report Date | Date | Date when the incident was initially reported | Format:  mm/dd/yyyy  WSDL name: initialRptDate |
| Damage Indicator | Char(1) | Indicates whether there as any damage to the property or not. | Values:   * Y * N   WSDL name: damageIndicator |
| Commercial Power Indicator | Char(1) | Indicates whether there is commercial power or not. | Values:   * Y * N   WSDL name: comPowerIndicator |
| On Generator Indicator | Char(1) | Indicates whether there the property is on a generator or not. | Values:   * Y * N   WSDL name: onGenIndicator |
| Un-Occupiable Indicator | Char(1) | Indicates whether there the property is occupiable or not. | Values:   * Y * N   WSDL name: unOccupiableIndicator |
| Mobility CO Indicator | Char(1) | Indicates whether there the property is a Mobility Central Office or not. | Values:   * Y * N   WSDL name: mobCOIndicator |
| Building Name | String(50) | Name of the building | WSDL name: buildingName |
| Building Address | String(255) | Address of the building | Screen name: Address  WSDL name: buildingAddress |
| State | String(2) | State associated with the location | WSDL name: state |
| Property Manager ATTUID | String(50) | AT&T User ID of the Property Manager. | WSDL name: PMAttuid |
| Building Status | Char(20) | Indicated the availability of the facility. | Values:   * OPENED * CLOSED * DELAYED OPENING * RE-OPENED   WSDL name: buildingStatus |
| Building Type | String(10) | Indicates the type of building | Values:   * ADM * CO * Others (see reference data section)   Screen name: Type  WSDL name: buildingType |
| CRE Lead | String(3) | Indicated the type of Corporate Real Estate individual leading the incident. | Values:   * PM * PDC   WSDL name: creLead |
| Estimated Expense Cost | Number | Estimated Expense Costs associated with the incident. | >= 0  WSDL name: estExpenseCost |
| Estimated Capital Cost | Number | Estimated Capital Costs associated with the incident. | >= 0  WSDL name: estCapCost |
| Electrical Issue Indicator | Char(1) | Indicates whether there is an electrical issue or not. | Values:   * Y * N   WSDL name: elecIssueIndicator |
| Electrical Issue Closed Indicator | Char(1) | Indicates whether the electrical issue was closed or not. | Values:   * Y * N   WSDL name: elecIssueClsdIndicator |
| Environmental Issue Indicator | Char(1) | Indicates whether there is an environmental issue or not. | Values:   * Y * N   WSDL name: envIssueIndicator |
| Environmental Issue Closed Indicator | Char(1) | Indicates whether the environmental issue was closed or not. | Values:   * Y * N   WSDL name: envIssueClsdIndicator |
| Fence/Gates Issue Indicator | Char(1) | Indicates whether there is a fence/gates issue or not. | Values:   * Y * N   WSDL name: fenceGateIssueIndicator |
| Fence/Gates Issue Closed Indicator | Char(1) | Indicates whether the fence/gates issue was closed or not. | Values:   * Y * N   WSDL name: fenceGateIssueIndicator |
| Generator Issue Indicator | Char(1) | Indicates whether there is a generator issue or not. | Values:   * Y * N   WSDL name: genIssueIndicator |
| Generator Issue Closed Indicator | Char(1) | Indicates whether the generator issue was closed or not. | Values:   * Y * N   WSDL name: genIssueClsdIndicator |
| Water Issue Indicator | Char(1) | Indicates whether there is a water issue or not. | Values:   * Y * N   WSDL name: waterIssueIndicator |
| Water Issue Closed Indicator | Char(1) | Indicates whether the water issue was closed or not. | Values:   * Y * N   WSDL name: waterIssueClsdIndicator |
| Grounds Issue Indicator | Char(1) | Indicates whether there is a grounds issue or not. | Values:   * Y * N   WSDL name: groundsIssueIndicator |
| Grounds Issue Closed Indicator | Char(1) | Indicates whether the grounds issue was closed or not. | Values:   * Y * N   WSDL name: groundsIssueClsdIndicator |
| Mechanical Issue Indicator | Char(1) | Indicates whether there is a mechanical issue or not. | Values:   * Y * N   WSDL name: mechIssueIndicator |
| Mechanical Issue Closed Indicator | Char(1) | Indicates whether the mechanical issue was closed or not. | Values:   * Y * N   WSDL name: mechIssueClsdIndicator |
| Other Issue Indicator | Char(1) | Indicates whether there is another issue or not. | Values:   * Y * N   WSDL name: otherIssueIndicator |
| Other Issue Closed Indicator | Char(1) | Indicates whether the “other” issue was closed or not. | Values:   * Y * N   WSDL name: otherIssueClsdIndicator |
| Plumbing Issue Indicator | Char(1) | Indicates whether there is a Plumbing issue or not. | Values:   * Y * N   WSDL name: plumbIssueIndicator |
| Plumbing Issue Closed Indicator | Char(1) | Indicates whether the Plumbing issue was closed or not. | Values:   * Y * N   WSDL name: plumbIssueClsdIndicator |
| Roofs Issue Indicator | Char(1) | Indicates whether there is a Roof issue or not. | Values:   * Y * N   WSDL name: roofsIssueIndicator |
| Roofs Issue Closed Indicator | Char(1) | Indicates whether the Roof issue was closed or not. | Values:   * Y * N   WSDL name: roofsIssueClsdIndicator |
| Safety Issue Indicator | Char(1) | Indicates whether there is a Safety issue or not. | Values:   * Y * N   WSDL name: safetyIssueIndicator |
| Safety Issue Closed Indicator | Char(1) | Indicates whether the Safety issue was closed or not. | Values:   * Y * N   WSDL name: safetyIssueClsdIndicator |
| Structural Issue Indicator | Char(1) | Indicates whether there is a Structural issue or not. | Values:   * Y * N   WSDL name: structIssueIndicator |
| Structural Issue Closed Indicator | Char(1) | Indicates whether the Structural issue was closed or not. | Values:   * Y * N   WSDL name: structIssueClsdIndicator |
| Assessment Notes | String (1000) | Notes providing an assessment of the incident. | WSDL name: assessNotes |
| Status Notes | String (1000) | Notes providing status of the incident. | WSDL name: statusNotes |
| Incident Notes | String (1000) | Notes/Comments about the incident | WSDL name: incidentNotes |

## Status Code definitions

|  |  |  |
| --- | --- | --- |
| Status Code | Description | Status Message |
| 000 | Success | SUCCESS |
| 001 | Access Denied | ACCESS DENIED |
| 002 | Session Timeout | SESSION TIMEOUT |
| 003 | Field Validation Error | [Error message] |
| 100 | No Incidents found | No Incidents found |
| 900 | DB Error | [Error message] |
| 901 | Application Error | [Error message] |
| 999 | Unexpected Error | [Error message] |

## Reference Data

The Mobile client will need the following reference data. It is expected that the mobile client application will include this as reference data (e.g. file).

| Data | Source Table | Notes |
| --- | --- | --- |
| Building Types | PROPTYPE | CRE Property Types |
| States | STATE | Static information: Can be included as a reference file included as part of the mobile application. |
| Event Names & Years | HURRICANE   * HURRNAME * YEAR | Events for prior, current and next year. |

## Sample SOAP requests/responses

1. **Login**

**Request:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:loginRequest>

<mes:userName>AA839G</mes:userName>

<mes:password>2b62ff9b70e704b9a36612bd0ea23927</mes:password>

</mes:loginRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<loginResponse xmlns="http://drt.att.com/drtservices/messages/">

<token>2XcI+9So8$7JMOnoNqjwA4DKw</token>

<authorization>ADM</authorization>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</loginResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **List Incidents**

**Request:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:listIncidentsRequest>

<mes:token>2XcI+9So8$7JMOnoNqjwA4DKw</mes:token>

<mes:geoLoc>77127</mes:geoLoc>

<!--Optional:-->

<mes:reqNumberTokens>?</mes:reqNumberTokens>

<!--Optional:-->

<mes:pageOffset>?</mes:pageOffset>

</mes:listIncidentsRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<listIncidentsResponse xmlns="http://drt.att.com/drtservices/messages/">

<incidentList>

<recNumber>4568</recNumber>

<geoLoc>77127</geoLoc>

<buildingName>OCEAN SPRINGS CO</buildingName>

<buildingAddress>1804 GOVERNMENT ST, OCEAN SPRINGS, MS</buildingAddress>

<eventName>Gustav</eventName>

<incidentStatus>Closed</incidentStatus>

<state>MS</state>

<pmATTUID>FOSTER, MELISSA</pmATTUID>

</incidentList>

<incidentList>

<recNumber>4573</recNumber>

<geoLoc>77127</geoLoc>

<buildingName>OCEAN SPRINGS CO</buildingName>

<buildingAddress>1804 GOVERNMENT ST, OCEAN SPRINGS, MS</buildingAddress>

<eventName>Gustav</eventName>

<incidentStatus>Open</incidentStatus>

<workReqNumber>080903021726</workReqNumber>

<state>MS</state>

<pmATTUID>FOSTER, MELISSA</pmATTUID>

</incidentList>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</listIncidentsResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Retrieve Incident:**

**Request:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:retrieveIncidentRequest>

<mes:token>2XcI+9So8$7JMOnoNqjwA4DKw</mes:token>

<mes:recNumber>4929</mes:recNumber>

</mes:retrieveIncidentRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<retrieveIncidentResponse xmlns="http://drt.att.com/drtservices/messages/">

<incidentStructure>

<geoLoc>27689</geoLoc>

<eventName>Ana</eventName>

<incidentStatus>Open</incidentStatus>

<incidentCompltnDate>2009-06-04</incidentCompltnDate>

<workReqNumber>112233445566778</workReqNumber>

<incidentYear>2009</incidentYear>

<reqATTUID>SJ0664</reqATTUID>

<contactPhone>205-669-2591</contactPhone>

<initialRptDate>2009-06-03</initialRptDate>

<damageIndicator>Y</damageIndicator>

<comPowerIndicator>N</comPowerIndicator>

<OnGeneratorIndicator>Y</OnGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

<mobCOIndicator>N</mobCOIndicator>

<buildingName>CHARLOTTE / RESEARCH DR -- REG DATA CTR</buildingName>

<buildingAddress>9139 RESEARCH DRIVE CHARLOTTE, NC</buildingAddress>

<state>NC</state>

<PMAttuid>JP6290</PMAttuid>

<buildingStatus>OPEN</buildingStatus>

<buildingType>DAT</buildingType>

<creLead>PM</creLead>

<estExpenseCost>80000</estExpenseCost>

<estCapCost>100000</estCapCost>

<elecIssueIndicator>Y</elecIssueIndicator>

<elecIssueClsdIndicator>N</elecIssueClsdIndicator>

<envIssueIndicator>N</envIssueIndicator>

<envIssueClsdIndicator>N</envIssueClsdIndicator>

<fenceGateIssueIndicator>N</fenceGateIssueIndicator>

<fenceGateIssueClsdIndicator>N</fenceGateIssueClsdIndicator>

<genIssueIndicator>N</genIssueIndicator>

<genIssueClsdIndicator>N</genIssueClsdIndicator>

<waterIssueIndicator>N</waterIssueIndicator>

<waterIssueClsdIndicator>N</waterIssueClsdIndicator>

<groundsIssueIndicator>N</groundsIssueIndicator>

<groundsIssueClsdIndicator>N</groundsIssueClsdIndicator>

<mechIssueIndicator>Y</mechIssueIndicator>

<mechIssueClsdIndicator>N</mechIssueClsdIndicator>

<otherIssueIndicator>N</otherIssueIndicator>

<otherIssueClsdIndicator>N</otherIssueClsdIndicator>

<plumbIssueIndicator>N</plumbIssueIndicator>

<plumbIssueClsdIndicator>N</plumbIssueClsdIndicator>

<roofsIssueIndicator>N</roofsIssueIndicator>

<roofsIssueClsdIndicator>N</roofsIssueClsdIndicator>

<safetyIssueIndicator>N</safetyIssueIndicator>

<safetyIssueClsdIndicator>N</safetyIssueClsdIndicator>

<structIssueIndicator>N</structIssueIndicator>

<structIssueClsdIndicator>N</structIssueClsdIndicator>

<assessNotes>Lightning knocked out commercial power, thus causing chiller motor to burn up.</assessNotes>

<statusNotes>Generators on. Backup chiller running; primary chiller off-line. Mechanical contractor in route.</statusNotes>

<incidentNotes>Cooling capacity running at 80% of normal.</incidentNotes>

<compltnDate>2009-06-04</compltnDate>

</incidentStructure>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</retrieveIncidentResponse>

</soapenv:Body>

</soapenv:Envelope>

Response with Pictures

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<retrieveIncidentResponse xmlns="http://drt.att.com/drtservices/messages/">

<incidentStructure>

<geoLoc>22222</geoLoc>

<eventName>TEST</eventName>

<incidentStatus>OPEN</incidentStatus>

<incidentCompltnDate>2011-06-27</incidentCompltnDate>

<workReqNumber>22222</workReqNumber>

<incidentYear>2011</incidentYear>

<reqATTUID>JW2479</reqATTUID>

<contactPhone>2222222222</contactPhone>

<initialRptDate>2011-06-27</initialRptDate>

<damageIndicator>N</damageIndicator>

<comPowerIndicator>N</comPowerIndicator>

<OnGeneratorIndicator>N</OnGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

<mobCOIndicator>N</mobCOIndicator>

<buildingName>TEST</buildingName>

<buildingAddress>TEST</buildingAddress>

<state>IL</state>

<PMAttuid>JW2479</PMAttuid>

<buildingStatus>OPEN</buildingStatus>

<buildingType>TEST</buildingType>

<creLead>AAA</creLead>

<estExpenseCost>0</estExpenseCost>

<estCapCost>0</estCapCost>

<elecIssueIndicator>N</elecIssueIndicator>

<elecIssueClsdIndicator>N</elecIssueClsdIndicator>

<envIssueIndicator>N</envIssueIndicator>

<envIssueClsdIndicator>N</envIssueClsdIndicator>

<fenceGateIssueIndicator>N</fenceGateIssueIndicator>

<fenceGateIssueClsdIndicator>N</fenceGateIssueClsdIndicator>

<genIssueIndicator>N</genIssueIndicator>

<genIssueClsdIndicator>N</genIssueClsdIndicator>

<waterIssueIndicator>N</waterIssueIndicator>

<waterIssueClsdIndicator>N</waterIssueClsdIndicator>

<groundsIssueIndicator>N</groundsIssueIndicator>

<groundsIssueClsdIndicator>N</groundsIssueClsdIndicator>

<mechIssueIndicator>N</mechIssueIndicator>

<mechIssueClsdIndicator>N</mechIssueClsdIndicator>

<otherIssueIndicator>N</otherIssueIndicator>

<otherIssueClsdIndicator>N</otherIssueClsdIndicator>

<plumbIssueIndicator>N</plumbIssueIndicator>

<plumbIssueClsdIndicator>N</plumbIssueClsdIndicator>

<roofsIssueIndicator>N</roofsIssueIndicator>

<roofsIssueClsdIndicator>N</roofsIssueClsdIndicator>

<safetyIssueIndicator>N</safetyIssueIndicator>

<safetyIssueClsdIndicator>N</safetyIssueClsdIndicator>

<structIssueIndicator>N</structIssueIndicator>

<structIssueClsdIndicator>N</structIssueClsdIndicator>

<assessNotes>N</assessNotes>

<statusNotes>N</statusNotes>

<incidentNotes>N</incidentNotes>

<compltnDate>2011-06-27</compltnDate>

</incidentStructure>

<imageURI>https://attcodingchallenge.att.com:82/101010/jona.jpg</imageURI>

<imageURI>https://attcodingchallenge.att.com:82/101010/machupicchu.jpg</imageURI>

<imageURI>https://attcodingchallenge.att.com:82/101010/val.jpg</imageURI>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</retrieveIncidentResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Update Incident**

**Request**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:updateIncidentRequest>

<mes:token>UbW$Qr+Ugc$RIhWvqjEqeTy8R</mes:token>

<mes:recNumber>4929</mes:recNumber>

<!--Optional:-->

<mes:incidentStructure>

<!--Optional:-->

<mes:geoLoc>27689</mes:geoLoc>

<!--Optional:-->

<mes:eventName>Anna</mes:eventName>

<!--Optional:-->

<mes:incidentStatus>Open</mes:incidentStatus>

<!--Optional:-->

<mes:incidentCompltnDate>2009-06-04</mes:incidentCompltnDate>

<!--Optional:-->

<mes:workReqNumber>112233445566778</mes:workReqNumber>

<!--Optional:-->

<mes:incidentYear>2009</mes:incidentYear>

<!--Optional:-->

<mes:reqATTUID>SJ0664</mes:reqATTUID>

<!--Optional:-->

<mes:contactPhone>2056692591</mes:contactPhone>

<!--Optional:-->

<mes:initialRptDate>2009-06-03</mes:initialRptDate>

<!--Optional:-->

<mes:damageIndicator>Y</mes:damageIndicator>

<!--Optional:-->

<mes:comPowerIndicator>N</mes:comPowerIndicator>

<!--Optional:-->

<mes:OnGeneratorIndicator>Y</mes:OnGeneratorIndicator>

<!--Optional:-->

<mes:unOccupiableIndicator>N</mes:unOccupiableIndicator>

<!--Optional:-->

<mes:mobCOIndicator>N</mes:mobCOIndicator>

<!--Optional:-->

<mes:buildingName>CHARLOTTE / RESEARCH DR -- REG DATA CTR</mes:buildingName>

<!--Optional:-->

<mes:buildingAddress>9139 RESEARCH DRIVE CHARLOTTE, NC</mes:buildingAddress>

<!--Optional:-->

<mes:state>NC</mes:state>

<!--Optional:-->

<mes:PMAttuid>JP6290</mes:PMAttuid>

<!--Optional:-->

<mes:buildingStatus>OPEN</mes:buildingStatus>

<!--Optional:-->

<mes:buildingType>DAT</mes:buildingType>

<!--Optional:-->

<mes:creLead>PM</mes:creLead>

<!--Optional:-->

<mes:estExpenseCost>80000</mes:estExpenseCost>

<!--Optional:-->

<mes:estCapCost>100000</mes:estCapCost>

<!--Optional:-->

<mes:elecIssueIndicator>Y</mes:elecIssueIndicator>

<!--Optional:-->

<mes:elecIssueClsdIndicator>N</mes:elecIssueClsdIndicator>

<!--Optional:-->

<mes:envIssueIndicator>N</mes:envIssueIndicator>

<!--Optional:-->

<mes:envIssueClsdIndicator>N</mes:envIssueClsdIndicator>

<!--Optional:-->

<mes:fenceGateIssueIndicator>N</mes:fenceGateIssueIndicator>

<!--Optional:-->

<mes:fenceGateIssueClsdIndicator>N</mes:fenceGateIssueClsdIndicator>

<!--Optional:-->

<mes:genIssueIndicator>N</mes:genIssueIndicator>

<!--Optional:-->

<mes:genIssueClsdIndicator>N</mes:genIssueClsdIndicator>

<!--Optional:-->

<mes:waterIssueIndicator>N</mes:waterIssueIndicator>

<!--Optional:-->

<mes:waterIssueClsdIndicator>N</mes:waterIssueClsdIndicator>

<!--Optional:-->

<mes:groundsIssueIndicator>N</mes:groundsIssueIndicator>

<!--Optional:-->

<mes:groundsIssueClsdIndicator>N</mes:groundsIssueClsdIndicator>

<!--Optional:-->

<mes:mechIssueIndicator>Y</mes:mechIssueIndicator>

<!--Optional:-->

<mes:mechIssueClsdIndicator>N</mes:mechIssueClsdIndicator>

<!--Optional:-->

<mes:otherIssueIndicator>N</mes:otherIssueIndicator>

<!--Optional:-->

<mes:otherIssueClsdIndicator>N</mes:otherIssueClsdIndicator>

<!--Optional:-->

<mes:plumbIssueIndicator>N</mes:plumbIssueIndicator>

<!--Optional:-->

<mes:plumbIssueClsdIndicator>N</mes:plumbIssueClsdIndicator>

<!--Optional:-->

<mes:roofsIssueIndicator>N</mes:roofsIssueIndicator>

<!--Optional:-->

<mes:roofsIssueClsdIndicator>N</mes:roofsIssueClsdIndicator>

<!--Optional:-->

<mes:safetyIssueIndicator>N</mes:safetyIssueIndicator>

<!--Optional:-->

<mes:safetyIssueClsdIndicator>N</mes:safetyIssueClsdIndicator>

<!--Optional:-->

<mes:structIssueIndicator>y</mes:structIssueIndicator>

<!--Optional:-->

<mes:structIssueClsdIndicator>y</mes:structIssueClsdIndicator>

<!--Optional:-->

<mes:assessNotes>Lightning knocked out commercial power, thus causing chiller motor to burn up.</mes:assessNotes>

<!--Optional:-->

<mes:statusNotes>Generators on. Backup chiller running; primary chiller off-line. Mechanical contractor in route.</mes:statusNotes>

<!--Optional:-->

<mes:incidentNotes>Cooling capacity running at 80% of normal.</mes:incidentNotes>

<!--Optional:-->

<mes:compltnDate>2009-06-10</mes:compltnDate>

</mes:incidentStructure>

</mes:updateIncidentRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<updateIncidentResponse xmlns="http://drt.att.com/drtservices/messages/">

<incidentStructure>

<geoLoc>27689</geoLoc>

<eventName>ANNA</eventName>

<incidentStatus>OPEN</incidentStatus>

<incidentCompltnDate>2009-06-10-05:00</incidentCompltnDate>

<workReqNumber>112233445566778</workReqNumber>

<incidentYear>2009</incidentYear>

<reqATTUID>SJ0664</reqATTUID>

<contactPhone>2056692591</contactPhone>

<initialRptDate>2009-06-03-05:00</initialRptDate>

<damageIndicator>Y</damageIndicator>

<comPowerIndicator>N</comPowerIndicator>

<OnGeneratorIndicator>Y</OnGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

<mobCOIndicator>N</mobCOIndicator>

<buildingName>CHARLOTTE / RESEARCH DR -- REG DATA CTR</buildingName>

<buildingAddress>9139 RESEARCH DRIVE CHARLOTTE, NC</buildingAddress>

<state>NC</state>

<PMAttuid>JP6290</PMAttuid>

<buildingStatus>OPEN</buildingStatus>

<buildingType>DAT</buildingType>

<creLead>PM</creLead>

<estExpenseCost>80000</estExpenseCost>

<estCapCost>100000</estCapCost>

<elecIssueIndicator>Y</elecIssueIndicator>

<elecIssueClsdIndicator>N</elecIssueClsdIndicator>

<envIssueIndicator>N</envIssueIndicator>

<envIssueClsdIndicator>N</envIssueClsdIndicator>

<fenceGateIssueIndicator>N</fenceGateIssueIndicator>

<fenceGateIssueClsdIndicator>N</fenceGateIssueClsdIndicator>

<genIssueIndicator>N</genIssueIndicator>

<genIssueClsdIndicator>N</genIssueClsdIndicator>

<waterIssueIndicator>N</waterIssueIndicator>

<waterIssueClsdIndicator>N</waterIssueClsdIndicator>

<groundsIssueIndicator>N</groundsIssueIndicator>

<groundsIssueClsdIndicator>N</groundsIssueClsdIndicator>

<mechIssueIndicator>Y</mechIssueIndicator>

<mechIssueClsdIndicator>N</mechIssueClsdIndicator>

<otherIssueIndicator>N</otherIssueIndicator>

<otherIssueClsdIndicator>N</otherIssueClsdIndicator>

<plumbIssueIndicator>N</plumbIssueIndicator>

<plumbIssueClsdIndicator>N</plumbIssueClsdIndicator>

<roofsIssueIndicator>N</roofsIssueIndicator>

<roofsIssueClsdIndicator>N</roofsIssueClsdIndicator>

<safetyIssueIndicator>N</safetyIssueIndicator>

<safetyIssueClsdIndicator>N</safetyIssueClsdIndicator>

<structIssueIndicator>y</structIssueIndicator>

<structIssueClsdIndicator>y</structIssueClsdIndicator>

<assessNotes>Lightning knocked out commercial power, thus causing chiller motor to burn up.</assessNotes>

<statusNotes>Generators on. Backup chiller running; primary chiller off-line. Mechanical contractor in route.</statusNotes>

<incidentNotes>Cooling capacity running at 80% of normal.</incidentNotes>

<compltnDate>2009-06-10-05:00</compltnDate>

</incidentStructure>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</updateIncidentResponse>

</soapenv:Body>

</soapenv:Envelope><soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<updateIncidentResponse xmlns="http://drt.att.com/drtservices/messages/">

<incidentStructure>

<geoLoc>27689</geoLoc>

<eventName>ANNA</eventName>

<incidentStatus>OPEN</incidentStatus>

<incidentCompltnDate>2009-06-10-05:00</incidentCompltnDate>

<workReqNumber>112233445566778</workReqNumber>

<incidentYear>2009</incidentYear>

<reqATTUID>SJ0664</reqATTUID>

<contactPhone>2056692591</contactPhone>

<initialRptDate>2009-06-03-05:00</initialRptDate>

<damageIndicator>Y</damageIndicator>

<comPowerIndicator>N</comPowerIndicator>

<OnGeneratorIndicator>Y</OnGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

<mobCOIndicator>N</mobCOIndicator>

<buildingName>CHARLOTTE / RESEARCH DR -- REG DATA CTR</buildingName>

<buildingAddress>9139 RESEARCH DRIVE CHARLOTTE, NC</buildingAddress>

<state>NC</state>

<PMAttuid>JP6290</PMAttuid>

<buildingStatus>OPEN</buildingStatus>

<buildingType>DAT</buildingType>

<creLead>PM</creLead>

<estExpenseCost>80000</estExpenseCost>

<estCapCost>100000</estCapCost>

<elecIssueIndicator>Y</elecIssueIndicator>

<elecIssueClsdIndicator>N</elecIssueClsdIndicator>

<envIssueIndicator>N</envIssueIndicator>

<envIssueClsdIndicator>N</envIssueClsdIndicator>

<fenceGateIssueIndicator>N</fenceGateIssueIndicator>

<fenceGateIssueClsdIndicator>N</fenceGateIssueClsdIndicator>

<genIssueIndicator>N</genIssueIndicator>

<genIssueClsdIndicator>N</genIssueClsdIndicator>

<waterIssueIndicator>N</waterIssueIndicator>

<waterIssueClsdIndicator>N</waterIssueClsdIndicator>

<groundsIssueIndicator>N</groundsIssueIndicator>

<groundsIssueClsdIndicator>N</groundsIssueClsdIndicator>

<mechIssueIndicator>Y</mechIssueIndicator>

<mechIssueClsdIndicator>N</mechIssueClsdIndicator>

<otherIssueIndicator>N</otherIssueIndicator>

<otherIssueClsdIndicator>N</otherIssueClsdIndicator>

<plumbIssueIndicator>N</plumbIssueIndicator>

<plumbIssueClsdIndicator>N</plumbIssueClsdIndicator>

<roofsIssueIndicator>N</roofsIssueIndicator>

<roofsIssueClsdIndicator>N</roofsIssueClsdIndicator>

<safetyIssueIndicator>N</safetyIssueIndicator>

<safetyIssueClsdIndicator>N</safetyIssueClsdIndicator>

<structIssueIndicator>y</structIssueIndicator>

<structIssueClsdIndicator>y</structIssueClsdIndicator>

<assessNotes>Lightning knocked out commercial power, thus causing chiller motor to burn up.</assessNotes>

<statusNotes>Generators on. Backup chiller running; primary chiller off-line. Mechanical contractor in route.</statusNotes>

<incidentNotes>Cooling capacity running at 80% of normal.</incidentNotes>

<compltnDate>2009-06-10-05:00</compltnDate>

</incidentStructure>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</updateIncidentResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Add Incident by Geoloc**

**Request**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:addIncidentRequest>

<mes:token>UbW$Qr+Ugc$RIhWvqjEqeTy8R</mes:token>

<!--Optional:-->

<mes:incidentStructure>

<!--Optional:-->

<mes:geoLoc>77127</mes:geoLoc>

<!--Optional:-->

<mes:eventName>Katrina</mes:eventName>

<!--Optional:-->

<mes:incidentStatus>Open</mes:incidentStatus>

<!--Optional:-->

<mes:incidentCompltnDate>2011-06-07</mes:incidentCompltnDate>

<!--Optional:-->

<mes:workReqNumber>080903021730</mes:workReqNumber>

<!--Optional:-->

<mes:incidentYear>2011</mes:incidentYear>

<!--Optional:-->

<mes:reqATTUID>VK0001</mes:reqATTUID>

<!--Optional:-->

<mes:contactPhone>8472224354</mes:contactPhone>

<!--Optional:-->

<mes:initialRptDate>2011-06-07</mes:initialRptDate>

<!--Optional:-->

<mes:damageIndicator>N</mes:damageIndicator>

<!--Optional:-->

<mes:comPowerIndicator>N</mes:comPowerIndicator>

<!--Optional:-->

<mes:OnGeneratorIndicator>N</mes:OnGeneratorIndicator>

<!--Optional:-->

<mes:unOccupiableIndicator>N</mes:unOccupiableIndicator>

<!--Optional:-->

<mes:mobCOIndicator>N</mes:mobCOIndicator>

<!--Optional:-->

<mes:buildingName>TRUMP TOWERS</mes:buildingName>

<!--Optional:-->

<mes:buildingAddress>1701 BLOOMINGSDALE DR, HOFFMAN ESTATES</mes:buildingAddress>

<!--Optional:-->

<mes:state>IL</mes:state>

<!--Optional:-->

<mes:PMAttuid>TJ8765</mes:PMAttuid>

<!--Optional:-->

<mes:buildingStatus>OPEN</mes:buildingStatus>

<!--Optional:-->

<mes:buildingType>DAT</mes:buildingType>

<!--Optional:-->

<mes:creLead>PM</mes:creLead>

<!--Optional:-->

<mes:estExpenseCost>20000</mes:estExpenseCost>

<!--Optional:-->

<mes:estCapCost>300000</mes:estCapCost>

<!--Optional:-->

<mes:elecIssueIndicator>N</mes:elecIssueIndicator>

<!--Optional:-->

<mes:elecIssueClsdIndicator>N</mes:elecIssueClsdIndicator>

<!--Optional:-->

<mes:envIssueIndicator>N</mes:envIssueIndicator>

<!--Optional:-->

<mes:envIssueClsdIndicator>Y</mes:envIssueClsdIndicator>

<!--Optional:-->

<mes:fenceGateIssueIndicator>Y</mes:fenceGateIssueIndicator>

<!--Optional:-->

<mes:fenceGateIssueClsdIndicator>Y</mes:fenceGateIssueClsdIndicator>

<!--Optional:-->

<mes:genIssueIndicator>Y</mes:genIssueIndicator>

<!--Optional:-->

<mes:genIssueClsdIndicator>Y</mes:genIssueClsdIndicator>

<!--Optional:-->

<mes:waterIssueIndicator>N</mes:waterIssueIndicator>

<!--Optional:-->

<mes:waterIssueClsdIndicator>N</mes:waterIssueClsdIndicator>

<!--Optional:-->

<mes:groundsIssueIndicator>N</mes:groundsIssueIndicator>

<!--Optional:-->

<mes:groundsIssueClsdIndicator>N</mes:groundsIssueClsdIndicator>

<!--Optional:-->

<mes:mechIssueIndicator>N</mes:mechIssueIndicator>

<!--Optional:-->

<mes:mechIssueClsdIndicator>N</mes:mechIssueClsdIndicator>

<!--Optional:-->

<mes:otherIssueIndicator>Y</mes:otherIssueIndicator>

<!--Optional:-->

<mes:otherIssueClsdIndicator>Y</mes:otherIssueClsdIndicator>

<!--Optional:-->

<mes:plumbIssueIndicator>N</mes:plumbIssueIndicator>

<!--Optional:-->

<mes:plumbIssueClsdIndicator>N</mes:plumbIssueClsdIndicator>

<!--Optional:-->

<mes:roofsIssueIndicator>N</mes:roofsIssueIndicator>

<!--Optional:-->

<mes:roofsIssueClsdIndicator>N</mes:roofsIssueClsdIndicator>

<!--Optional:-->

<mes:safetyIssueIndicator>N</mes:safetyIssueIndicator>

<!--Optional:-->

<mes:safetyIssueClsdIndicator>N</mes:safetyIssueClsdIndicator>

<!--Optional:-->

<mes:structIssueIndicator>N</mes:structIssueIndicator>

<!--Optional:-->

<mes:structIssueClsdIndicator>N</mes:structIssueClsdIndicator>

<!--Optional:-->

<mes:assessNotes>Storm caused major damage.</mes:assessNotes>

<!--Optional:-->

<mes:statusNotes>Repair work continuing.</mes:statusNotes>

<!--Optional:-->

<mes:incidentNotes>Temp too high.</mes:incidentNotes>

<!--Optional:-->

<mes:compltnDate>2011-06-07</mes:compltnDate>

</mes:incidentStructure>

</mes:addIncidentRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<addIncidentResponse xmlns="http://drt.att.com/drtservices/messages/">

<incidentStructure>

<geoLoc>77127</geoLoc>

<eventName>KATRINA</eventName>

<incidentStatus>OPEN</incidentStatus>

<incidentCompltnDate>2011-06-07-05:00</incidentCompltnDate>

<workReqNumber>080903021730</workReqNumber>

<incidentYear>2011</incidentYear>

<reqATTUID>VK0001</reqATTUID>

<contactPhone>8472224354</contactPhone>

<initialRptDate>2011-06-07-05:00</initialRptDate>

<damageIndicator>N</damageIndicator>

<comPowerIndicator>N</comPowerIndicator>

<OnGeneratorIndicator>N</OnGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

<mobCOIndicator>N</mobCOIndicator>

<buildingName>TRUMP TOWERS</buildingName>

<buildingAddress>1701 BLOOMINGSDALE DR, HOFFMAN ESTATES</buildingAddress>

<state>IL</state>

<PMAttuid>TJ8765</PMAttuid>

<buildingStatus>OPEN</buildingStatus>

<buildingType>DAT</buildingType>

<creLead>PM</creLead>

<estExpenseCost>20000</estExpenseCost>

<estCapCost>300000</estCapCost>

<elecIssueIndicator>N</elecIssueIndicator>

<elecIssueClsdIndicator>N</elecIssueClsdIndicator>

<envIssueIndicator>N</envIssueIndicator>

<envIssueClsdIndicator>Y</envIssueClsdIndicator>

<fenceGateIssueIndicator>Y</fenceGateIssueIndicator>

<fenceGateIssueClsdIndicator>Y</fenceGateIssueClsdIndicator>

<genIssueIndicator>Y</genIssueIndicator>

<genIssueClsdIndicator>Y</genIssueClsdIndicator>

<waterIssueIndicator>N</waterIssueIndicator>

<waterIssueClsdIndicator>N</waterIssueClsdIndicator>

<groundsIssueIndicator>N</groundsIssueIndicator>

<groundsIssueClsdIndicator>N</groundsIssueClsdIndicator>

<mechIssueIndicator>N</mechIssueIndicator>

<mechIssueClsdIndicator>N</mechIssueClsdIndicator>

<otherIssueIndicator>Y</otherIssueIndicator>

<otherIssueClsdIndicator>Y</otherIssueClsdIndicator>

<plumbIssueIndicator>N</plumbIssueIndicator>

<plumbIssueClsdIndicator>N</plumbIssueClsdIndicator>

<roofsIssueIndicator>N</roofsIssueIndicator>

<roofsIssueClsdIndicator>N</roofsIssueClsdIndicator>

<safetyIssueIndicator>N</safetyIssueIndicator>

<safetyIssueClsdIndicator>N</safetyIssueClsdIndicator>

<structIssueIndicator>N</structIssueIndicator>

<structIssueClsdIndicator>N</structIssueClsdIndicator>

<assessNotes>Storm caused major damage.</assessNotes>

<statusNotes>Repair work continuing.</statusNotes>

<incidentNotes>Temp too high.</incidentNotes>

<compltnDate>2011-06-07-05:00</compltnDate>

</incidentStructure>

<recNumber>101011</recNumber>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>Success</statusMessage>

</requestStatus>

</addIncidentResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Get Admin Condition Report**

**Request:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:getAdminConditionRptRequest>

<mes:token>UbW$Qr+Ugc$RIhWvqjEqeTy8R</mes:token>

<!--Optional:-->

<mes:year>2011</mes:year>

<!--Optional:-->

<mes:eventName>Alex</mes:eventName>

</mes:getAdminConditionRptRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<getAdminConditionRptResponse xmlns="http://drt.att.com/drtservices/messages/">

<reportList>

<recNumber>10102</recNumber>

<geoLoc>10006</geoLoc>

<incidentYear>2011</incidentYear>

<buildingName>10006-BIRMINGHAM / HWY 280 - AOC -- OFCA</buildingName>

<buildingAddress>3196 HWY 280 E BIRMINGHAM, AL</buildingAddress>

<eventName>Alex</eventName>

<state>AL</state>

<pmATTUID>GS7675</pmATTUID>

<assessNotes>Test</assessNotes>

<damageIndicator>Y</damageIndicator>

<comPowerIndicator>N</comPowerIndicator>

<onGeneratorIndicator>N</onGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

</reportList>

<reportList>

<recNumber>22222</recNumber>

<geoLoc>10006</geoLoc>

<incidentYear>2011</incidentYear>

<buildingName>10006-BIRMINGHAM / HWY 280 - AOC -- OFCA</buildingName>

<buildingAddress>3196 HWY 280 E BIRMINGHAM, AL</buildingAddress>

<eventName>Alex</eventName>

<state>AL</state>

<pmATTUID>GS7675</pmATTUID>

<assessNotes>Test ATT Coding Challenge</assessNotes>

<statusNotes>Test</statusNotes>

<damageIndicator>Y</damageIndicator>

<comPowerIndicator>Y</comPowerIndicator>

<onGeneratorIndicator>Y</onGeneratorIndicator>

<unOccupiableIndicator>N</unOccupiableIndicator>

</reportList>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>SUCCESS.</statusMessage>

</requestStatus>

</getAdminConditionRptResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Get CRE Report**

**Request:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:getCREClosureDelayedRptRequest>

<mes:token>UbW$Qr+Ugc$RIhWvqjEqeTy8R</mes:token>

<!--Optional:-->

<mes:year>2011</mes:year>

<!--Optional:-->

<mes:eventName>Alex</mes:eventName>

</mes:getCREClosureDelayedRptRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<getCREClosureDelayedRptResponse xmlns="http://drt.att.com/drtservices/messages/">

<reportList>

<recNumber>10102</recNumber>

<geoLoc>10006</geoLoc>

<initialRptDate>2011-03-10</initialRptDate>

<buildingStatus>DELAYED OPENING</buildingStatus>

<buildingType>ADM</buildingType>

<buildingAddress>3196 HWY 280 E BIRMINGHAM, AL</buildingAddress>

<state>AL</state>

<pmATTUID>GS7675</pmATTUID>

<assessNotes>Test</assessNotes>

</reportList>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>SUCCESS.</statusMessage>

</requestStatus>

</getCREClosureDelayedRptResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Add Incident Pic**

**Request: (Including HTTP header fields):**

POST /DRTWeb/DRTServices HTTP/1.1

Accept-Encoding: gzip,deflate

➊

SOAPAction: " /DRTWeb/DRTServices"

Content-Type: multipart/related; type="application/xop+xml"; start="<rootpart@soapui.org>"; start-info="text/xml"; boundary="**MIME\_boundary**"

MIME-Version: 1.0

Content-Length: 3527

**--MIME\_boundary**

➋

Content-Type: application/xop+xml; charset=UTF-8; type="text/xml" \_\_

Content-Transfer-Encoding: 8bit\_\_

Content-ID: <rootpart@soapui.org>\_\_

\_\_

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:addIncidentPicRequest>

<mes:token>SjJyeBT3BLGd4bgAGdfl0qLwL</mes:token>

<mes:recNumber>5678</mes:recNumber>

<mes:fileName>image001.jpg</mes:fileName>

<mes:picture><inc:Include href="**cid:val.jpg**" xmlns:inc="http://www.w3.org/2004/08/xop/include"/></mes:picture>

</mes:addIncidentPicRequest>

</soapenv:Body>

</soapenv:Envelope>

**--MIME\_boundary**

Content-Type: image/jpeg; name=val.jpg\_\_

Content-Transfer-Encoding: binary\_\_

Content-ID: <**val.jpg**>\_\_

Content-Disposition: attachment; name="val.jpg"; filename="val.jpg"\_\_

\_\_

**…Raw Binary JPG image (this is the actual content of image file)…**

➌

**--MIME\_boundary--**

Note that the example above includes all of the HTTP headers for the “add Incident Picture” operation as an illustration. Please refer to the “SOAP with attachment” specification link provided on the requirements section for more information. If you decide to generate this request manually (without the use of a convenience library), then consider the following elements of the request that have been annotated.

➊ HTTP header section (main). They can be generated using the Apache setHeader method. Make sure that all of the headers (except Content-Length) are set to the values stated on this example. Note that the Content-Length field is automatically generated by the Apache HTTP library. In addition, note that first MIME boundary is not part of the main header section.

➋ Sub HTTP headers section supporting each part of the message. Note that the \_\_ represent a carriage return and a new line character (“\r\n”). These two delimiters are required by the HTTP specification for correct parsing of the message. Also consider the **MIME\_boundary** element and the - - at the beginning of each part of the message.

➌ Image file portion. This is the raw (binary) content of the JPEG image file. Given that the content of this part is in binary format, consider an appropriate entity (e.g. ByteArrayEntity) to encode your entire payload (i.e. image file + SOAP xml request). This will ensure that the server is capable of reading and storing the image file accordingly. Also note the closing **MIME\_boundary** element.

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<addIncidentPicResponse xmlns="http://drt.att.com/drtservices/messages/">

<imageURI>https://attcodingchallenge.att.com:82/5678/val.jpg</imageURI>

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>SUCCESS.</statusMessage>

</requestStatus>

</addIncidentPicResponse>

</soapenv:Body>

</soapenv:Envelope>

1. **Delete Incident PicRequest:**

**<soapenv:Envelope x**mlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mes="http://drt.att.com/drtservices/messages/">

<soapenv:Header/>

<soapenv:Body>

<mes:deleteIncidentPicRequest>

<mes:token>UbW$Qr+Ugc$RIhWvqjEqeTy8R</mes:token>

<!--Optional:-->

<mes:recNumber>101040</mes:recNumber>

<!--Optional:-->

<mes:fileName>val.jpg</mes:fileName>

</mes:deleteIncidentPicRequest>

</soapenv:Body>

</soapenv:Envelope>

**Response:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">

<soapenv:Body>

<deleteIncidentPicResponse xmlns="http://drt.att.com/drtservices/messages/">

<requestStatus>

<statusCode>000</statusCode>

<statusMessage>SUCCESS.</statusMessage>

</requestStatus>

</deleteIncidentPicResponse>

</soapenv:Body>

</soapenv:Envelope>

# Appendix

## Glossary

|  |  |
| --- | --- |
| Field | Description |
| **ADDRESS** | Street and House Number for Building |
| **BLDG\_STATE** | Building State - Open, Closed, Re-Opened or Delayed Opening |
| **Building Name** | Building name prepended with GeoLoc |
| **City** | City for Building |
| **CRE** | Corporate Real Estate |
| **CRE\_LEAD** | Corporate Real Estate Lead (PM or D&C - Design and Construction) |
| **DRT** | Damage Recovery Assessment Tool |
| **DUE\_DATE** | Due Date |
| **FM/PM** | Facility Manager or Property Manager |
| **GLC/GEOLOC** | Geographic Location Code assigned to the Building |
| **IR\_DATE** | Initial Request Date and Time |
| **LONGADDR** | Full Address - House #, Street, City, St and Zip |
| **OPEN\_CLOSED** | Ticket is Open or Closed |
| **PROPTYPE** | Property Type (CO - Central Office, ADM - Administration Building, WCR - Work Center, WAR - Warehouse, RET - Retail Store, TWR - Tower) |
| **REC\_CREATE** | Record Created By |
| **REC\_CREATE\_DATE** | Record Created Date and Time |
| **REC\_MODIFY** | Record Modified By |
| **REC\_MODIFY\_DATE** | Record Modified Date and Time |
| **REQUESTOR** | ATTUID of Person putting in the record/ticket |
| **ST** | State for Building |
| **STORMNAME** | Hurricane Name or Storm type |
| **WR** | Work Request # - Used if repairs are completed by a contractor/vendor |
| **YEAR** | Storm Year |