|  |
| --- |
| **Philly 311 CRM**  **Technical Design – PublicStuff Integration**  **V1.3** |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision Number** | **Revision Date** | **Summary of Changes** | **Author** |
| Draft | 02/04/2014 | Created Draft | Prabhakar/Murugaboopathi |
| 1.0 | 02/13/2014 | Architecture Diagram and Mapping | Prabhakar/Murugaboopathi |
| 1.1 | 02/16/2014 | Update to API Descriptions and other sections | Steve Waters |
| 1.2 | 23/01/2015 | Update to field mappings and record type mappings | Hari Krishna Kousik Ogirala/ Sreelatha SK |
| 1.3 | 28/05/2015 | Changes w.r.t 09158078 for handling the PS related data errors (field mapping) in Integration and SF, move the requests into “311 Mobile Requests” queue | Ashritha S/ Sreelatha SK |

[1 Introduction 5](#_Toc420571022)

[1.1 Document Purpose 5](#_Toc420571023)

[1.2 System Overview 5](#_Toc420571024)

[2 PublicStuff System Integration 6](#_Toc420571025)

[2.1 OVERVIEW 6](#_Toc420571026)

[2.2 CONFIGURATION 6](#_Toc420571027)

[2.3 Inbound Data Flow 7](#_Toc420571028)

[2.3.1 Service Request Submission 7](#_Toc420571029)

[2.3.1.1Request Submit Service API 7](#_Toc420571030)

[2.3.1.3 Implemented By 8](#_Toc420571031)

[2.3.1.4 URL 8](#_Toc420571032)

[2.3.1.5 Parameters 8](#_Toc420571033)

[2.3.1.6 PublicStuff Request Type to SFDC Record Type Mapping 10](#_Toc420571034)

[2.3.1.7 Returns 11](#_Toc420571035)

[2.3.1.8 Implementation Notes 11](#_Toc420571036)

[2.3.1.9 Authentication 11](#_Toc420571037)

[2.3.1.10 Salesforce Web Service API 11](#_Toc420571038)

[2.3.2 Service Request Comment Submission 11](#_Toc420571039)

[2.3.2.1Request Submit Service API 11](#_Toc420571040)

[2.3.2.2 Implemented By 11](#_Toc420571041)

[2.3.2.3 URL 12](#_Toc420571042)

[2.3.2.4 Parameters 12](#_Toc420571043)

[2.3.2.5 Returns 12](#_Toc420571044)

[2.3.2.7 Implementation Notes 12](#_Toc420571045)

[2.3.2.8 Authentication 13](#_Toc420571046)

[2.3.2.9 Salesforce Web Service API 13](#_Toc420571047)

[2.4 Outbound Data Flow 13](#_Toc420571048)

[2.4.1 Case Status Update 13](#_Toc420571049)

[2.4.1.1 Case Status Update Outbound Message 13](#_Toc420571050)

[2.4.1.2 Fields Contained in message 13](#_Toc420571051)

[2.4.1.3 Request Status Update API 14](#_Toc420571052)

[2.4.1.4 Implemented By 14](#_Toc420571053)

[2.4.1.5 URL 14](#_Toc420571054)

[2.4.1.6 Parameters 14](#_Toc420571055)

[2.4.1.7 Response 14](#_Toc420571056)

[2.4.1.8 Implementation Notes 15](#_Toc420571057)

[2.4.1.9 Authentication 15](#_Toc420571058)

[2.4.2 Case Comment Submit 15](#_Toc420571059)

[2.4.2.1 Case Comment Create Outbound Message 15](#_Toc420571060)

[2.4.2.2 Triggering Workflow 15](#_Toc420571061)

[2.4.2.3 Fields Contained in message 15](#_Toc420571062)

[2.4.2.4 Case Comment API 16](#_Toc420571063)

[2.4.2.5 Implemented By 16](#_Toc420571064)

[2.4.2.6 URL 16](#_Toc420571065)

[2.4.2.7 Parameters 16](#_Toc420571066)

[2.4.2.8 Response 16](#_Toc420571067)

[2.4.2.9 Schedule 16](#_Toc420571068)

[2.4.2.10 Authentication 17](#_Toc420571069)

[2.5 LOGGING AND EXCEPTION HANDLING 17](#_Toc420571070)

[2.5.1 Logging 17](#_Toc420571071)

[2.5.2 Exceptions 17](#_Toc420571072)

[2.6 PUBLICSTUFF-PHILLY 311 CRM ENTITY FIELDS MAPPING 17](#_Toc420571073)

[Request Specific Fields and Their Mappings 18](#_Toc420571074)

[2.6 DEVELOPMENT 20](#_Toc420571075)

[2.6.1 Coding Standards 20](#_Toc420571076)

[2.6.2 Developers 20](#_Toc420571077)

[2.6.3 Version Control 20](#_Toc420571078)

# 1 Introduction

## 1.1 Document Purpose

The purpose of this document is to provide technical design details needed to successfully implement PublicStuff integration for Philly311 CRM.

|  |
| --- |
| 1.2 System Overview |

Philly311 CRM solution provides 311 Agents, 311 Supervisors, Managing Directors, Departmental SMEs, Department Leadership, and City Leadership to leverage Salesforce.com functionality to engage and provide new outreach opportunities with the Community Partners. Philly311 CRM provides ability to look Customer info, Knowledge Base to help Agents, Maintain Service Requests, and generate analytical reporting to support decision making process, establish more Channels (using Social Media) and collaborate using Chatter.

# 2 PublicStuff System Integration

## 2.1 OVERVIEW

The design solution will provide seamless bi-directional integration to the City’s PublicStuff account. Salesforce Service Cloud application will be configured to integrate with PublicStuff using Software AG Integration Server hosted within City's premises. Web Services developed in and exposed by the Integration Server will accept incoming data from PublicStuff Mobile application and website. Integration Server Services will be developed to push data from the City to the PublicStuff mobile application and Website.

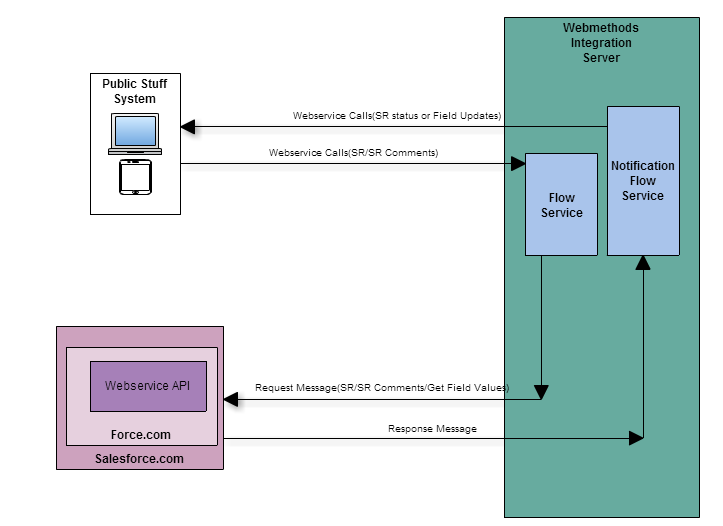


Diagram 1.1 – Architecture of PublicStuff integration

## 2.2 CONFIGURATION

All communication between systems is done using standard HTTP/HTTPS based services – REST and SOAP.

All communication between the Integration Server and PublicStuff will be done using REST interfaces. Inbound REST interfaces are built in feature of Integration Server. Outbound REST interfaces are called using standard HTTP/HTTPS callouts (pub.client:http) and JSON parsing capabilities (pub.json:jsonStringToDocument, pub.json:documentToJSONString) in the Integration Server.

All communication between the Integration Server and Salesforce will be done using SOAP interfaces. Both Inbound and Outbound SOAP interfaces are built in features of Integration Server.

The following sections provide a detailed description of the Inbound and Outbound data flow between PublicStuff, Integration Server and Salesforce.

## 2.3 Inbound Data Flow

Inbound messages from PublicStuff will be new Service Requests submitted by PublicStuff users and Service Request Comments added by users. The process to transmit these requests to the City will be as follows:

* PublicStuff will continue to use their batch process which will be scheduled in PublicStuff System to process requests in near real-time.
* PublicStuff will call web services exposed by the Integration Server to submit service request and comments for request types that *are* part of the 311 system. This new Integration Server web service (described below) will create/ update the associated Salesforce Case or add Case Comments using the standard Salesforce Web Service API and return the Case Number to the caller. Error messages will be returned on failure.
* PublicStuff will continue to validate user entered addresses using the City’s GIS services.
* Integration Server web service will call the City’s ULRS services to determine service request specific GIS information.
* ~~PublicStuff will process request types that~~ *~~are not~~* ~~part of the 311 system pilot as they do today to support the existing processes (NOVO/CityWorks).~~
* PublicStuff will build in a configurable mechanism to route request types to new process as these request type come online in the new system (Salesforce).
* All the request types routed to new system that is salesforce will map to 311 Mobile Requests queue in case of data error.

### Service Request Submission

The following API calls will be used/implemented to support request submission.

#### 2.3.1.1Request Submit Service API

This is an Integration Server REST service that will insert a case into the 311 CRM.

#### 2.3.1.3 Implemented By

Philly311 (Unisys)

#### 2.3.1.4 URL

POST /rest/philly311/request\_submit

Example:

|  |
| --- |
| /rest/philly311/request\_submit?request\_id=99999&request\_type\_name=pothole+repair&…. |

#### Parameters

POST parameters and their mapping to Salesforce fields are listing in the following table.

| **Parameter** | **Data Type** | **Description** | **Map to** | **Req’d** |
| --- | --- | --- | --- | --- |
| request\_id | string | PS request Id | Case.CustomerRequestId\_\_c | Y |
| request\_type\_name | string |  | Case.CustomerRequestType\_\_c and Case.RecordType.Name via mapping below | Y |
| title | string |  | Case.Subject | Y |
| request\_status | string |  | Case.Status | N |
| description | string | This contains additional Service Request specific fields added as additional post parameters, along with description.  Example: *Test Description*  Custom fields:  Is the issue in the front or rear?: Front Your Name & Contact Number: Test Name | Case.Description | N |
| address | string |  | Case.Street\_\_c | N |
| location | string | Parse out city and state. Will be overwritten if zip code is supplied | Case.City\_\_c, Case.State\_\_c | N |
| lat | decimal |  | TBD | N |
| lng | decimal |  | TBD | N |
| zipcode | string |  | Case.Zip\_\_c | N |
| channel | string | Mobile, Web, SMS | Case.Source | N |
| image | string (URL) |  | Case.Customer\_Supplied\_Image\_\_c | N |
| cust\_first\_name | string | Related contact | Contact.FirstName | N |
| cust\_last\_name | string |  | Contact.LastName | N |
| cust\_address | string |  | Contact.MailingAddress | N |
| cust\_zip | string |  | Contact.MailingPostalCode | N |
| cust\_phone | string |  | Contact.Phone | N |
| cust\_email | string |  | Contact.Email | N |
| ~~request specific fields~~ |  | ~~Additional request specific fields shall be added as additional post parameters.~~ | ~~Appropriate Case Field~~ | ~~N~~ |

#### PublicStuff Request Type to SFDC Record Type Mapping

|  |  |
| --- | --- |
| **PublicStuff Request Type Name** | **SFDC RecordType** |
| Abandoned Automobile | Abandoned Vehicle |
| Dangerous Building | Building Dangerous |
| Fallen Tree | Street Trees |
| Graffiti Removal | Graffiti Removal |
| Illegal Dumping | Illegal Dumping |
| Maintenance Commercial | Maintenance Residential or Commercial |
| Maintenance Residential | Maintenance Residential or Commercial |
| Other | Miscellaneous |
| Park Conditions (Safety & Maintenance) | Parks and Rec Safety and Maintenance |
| Pothole Repair | Street Defect |
| Recycling Collection | Rubbish/Recyclable Material Collection |
| Rubbish Collection | Rubbish/Recyclable Material Collection |
| Street Light Outage | Street Light Outage |
| Vacant Commercial | Vacant House or Commercial |
| Vacant House | Vacant House or Commercial |
| Vacant Lot Clean-up | Vacant Lot Clean-Up |

#### Returns

The service returns the following as a JSON string:

Example:

|  |
| --- |
| {  "case\_id": "5001600000d6J76AAE",  "case\_number": "0010000001",  "message" : null  } |

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Data Type** | **Description** | **Map to** |
| case\_id | string | Salesforce case ID, null on error |  |
| case\_number | string | Salesforce Case Number, null on error | Case.CaseNumber |
| Message | string | Error message describing service failure, ‘OK’ on success |  |

#### 2.3.1.8 Implementation Notes

Batches sent approximately every 10 minutes from PublicStuff

#### 2.3.1.9 Authentication

Basic Authentication with Username/Password supplied by City

#### 2.3.1.10 Salesforce Web Service API

See <http://www.salesforce.com/us/developer/docs/api/index_Left.htm>

### Service Request Comment Submission

The following API calls will be used/implemented to support request comment submission

#### 2.3.2.1Request Submit Service API

This is an Integration Server REST service that will insert a case comment into the 311 CRM

#### 2.3.2.2 Implemented By

Philly311 (Unisys)

#### 2.3.2.3 URL

POST /rest/philly311/comment\_submit

Example:

|  |
| --- |
| / rest/philly311/comment\_submit?request\_id=99999& comment=hello+world&is\_private=0 |

#### 2.3.2.4 Parameters

POST parameters and their mapping to Salesforce fields are listing in the following table.

| **Parameter** | **Data Type** | **Description** | **Map to** | **Req’d** |
| --- | --- | --- | --- | --- |
| request\_id | string | PS request Id | CaseComment.ParentId | Y |
| Comment | string |  | CaseComment.CommentBody | Y |
| is\_private | string |  | CaseComment.IsPublished | N |

#### 2.3.2.5 Returns

The service returns the following as a JSON string:

Example:

|  |
| --- |
| {  "case\_comment\_id": "00d0000000000000000",  "Message" : null  } |

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Data Type** | **Description** | **Map to** |
| case\_comment\_id | string | Salesforce Case Comment Id, null on error | CaseComment.Id |
| Error | string | Error message describing service failure, null on success |  |

#### 2.3.2.7 Implementation Notes

Batches sent approximately every 10 minutes from PublicStuff

#### 2.3.2.8 Authentication

Basic Authentication with Username/Password supplied by City

#### 2.3.2.9 Salesforce Web Service API

See <http://www.salesforce.com/us/developer/docs/api/index_Left.htm>

## 2.4 Outbound Data Flow

Outbound data flow from Salesforce to PublicStuff will be Service Request status updates in response to Case status changes in Salesforce and Service Request Comment submission in response to Case Comment creation in Salesforce.

The outbound flow will be triggered using the Salesforce Outbound Messaging feature. Salesforce Outbound Messaging allows us to specify that changes to fields within Salesforce can cause messages with field values to be sent to designated external servers via SOAP message. Integration Server web services will be developed to receive these messages and propagate those changes to integrated systems (PublicStuff, CityWorks, etc.). Salesforce Outbound Messages are initiated by workflows within Salesforce. Workflows will be created to detect Case status changes and creation of new Case Comments. A description of the Outbound Messaging SOAP message structure can be found at <http://www.salesforce.com/us/developer/docs/api/Content/sforce_api_om_outboundmessaging.htm>

### 2.4.1 Case Status Update

For Case status changes, Salesforce will call an Integration Server Web Services with the new status, which will then use PublicStuff Request Status Update API call to update the Service Request status in PublicStuff.

#### 2.4.1.1 Case Status Update Outbound Message

Standard Salesforce Outbound SOAP message. Interface implemented by Integration Server.

#### 2.4.1.2 Fields Contained in message

From Case object:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Data Type** | **Description** |
| Id | string | Salesforce Id |
| CaseNumber | string |  |
| CustomerRequestId\_\_c | string | PS request Id |
| Status | string |  |

#### 2.4.1.3 Request Status Update API

A REST service that will update the status on a request

#### 2.4.1.4 Implemented By

PublicStuff standard (hidden) API call

#### 2.4.1.5 URL

POST /api/2.0/request\_update

#### 2.4.1.6 Parameters

Example.

|  |
| --- |
| /api/2.0/request\_update?return\_type=json&request\_id=303400&status=Closed&api\_key=d950284159de2fdc502b584685681bd6 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Data Type** | **Description** | **Map to** |
| request\_id | string | PS request Id | Case.CustomerRequestId\_\_c |
| Status | string |  | Case.RecordTypeId |
| api\_key | string | City’s API Key. Stored in settings |  |
| return\_type | string | “JSON” |  |

#### 2.4.1.7 Response

Standard PublicStuff status JSON response

Example

|  |
| --- |
| {  "response": {  "status": {  "type": "success",  "message": "Success",  "code": 200,  "code\_message": "Ok"  }  }  } |

#### 2.4.1.8 Implementation Notes

Process will be driven by Outbound Message in Salesforce

#### 2.4.1.9 Authentication

API Key parameter

### 2.4.2 Case Comment Submit

For Case Comment creation, Salesforce will call an Integration Server web services with a reference to the parent case and the contents of the comment, which will then use PublicStuff Case Comment API call to add the comment to the Service Request in PublicStuff.

#### 2.4.2.1 Case Comment Create Outbound Message

Standard Salesforce Outbound SOAP message. Interface implemented by Integration Server.

#### 2.4.2.2 Triggering Workflow

A workflow will be created that will trigger the Outbound message when a Case Comment is created for a PublicStuff Case (Case Customer Request Id != null).

#### 2.4.2.3 Fields Contained in message

From Case Comment object:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Data Type** | **Description** |
| Id | string | Salesforce Id |
| ParentId | string | Parent Case Id, will be used to lookup CustomerRequestId\_\_c (request\_id) |
| CommentBody | string |  |
| IsPublished | string | Will be used to set PublicStuff is\_private flag. |

#### 2.4.2.4 Case Comment API

A REST service that will add comments to a request

#### 2.4.2.5 Implemented By

PublicStuff standard API call

#### 2.4.2.6 URL

POST /api/2.0/comments\_submit

#### 2.4.2.7 Parameters

Example

|  |
| --- |
| /api/2.0/comment\_submit?return\_type=json&request\_id=303400&comment=comment&is\_private=0&api\_key=d950284159de2fdc502b584685681bd6 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Data Type** | **Description** | **Map to** |
| request\_id | string | PS request Id | CaseComment.Parent.CustomerRequestId\_\_c |
| Comment | string |  | CaseComment.CommentBody |
| is\_private | string | Marks the comment as public(1) or private(0). Private requests are not visible to the public. | CaseComment.IsPublished (not) |
| api\_key | string | City’s API Key. Stored in settings |  |
| return\_type | string | “JSON” |  |

#### 2.4.2.8 Response

Standard PublicStuff status JSON response. See above for example.

#### 2.4.2.9 Schedule

Process will be driven by Outbound Message in Salesforce.

#### 2.4.2.10 Authentication

API Key parameter

## 2.5 LOGGING AND EXCEPTION HANDLING

### 2.5.1 Logging

1. Audit Log: To monitor individual adapter services using the audit log.
2. Error Log: The Integration Server services automatically posts fatal-level and error-level log messages to the error log.
3. Server Log: The Integration Server services post messages to the server log, depending on how the server log is configured.

* Fatal-level through debug-level log messages appear as Adapter Runtime log messages.
* Trace-level log messages appear as Salesforce.com Adapter log messages.

### 2.5.2 Exceptions

Exceptions raised by the Integration Server services will contain the SOAP fault errors that Salesforce.com returns in its error stack.

The Force.com Web services API calls provides the following types of error handling:

* + For errors resulting from badly formed messages, failed authentication, or similar problems, the Force.com Web services API returns a SOAP fault message with an associated Exception Code.
  + For most calls, if the error occurs because of a problem specific to the query, the API returns an Error. For example, if a create request contains more than 200 objects, the API returns an Error.

## 2.6 PUBLICSTUFF-PHILLY 311 CRM ENTITY FIELDS MAPPING

Fields in the PublicStuff system are mapped to Salesforce.com Case and Case Comment fields as described below.

| **Public Stuff Entity Fields** | **Data Type** | **Map to Salesforce Case Object fields** |
| --- | --- | --- |
| request\_id | String | Case.CustomerRequestId\_\_c |
| request\_type\_name | String | Case.RecordType.Name (may change) |
| Title | String | Case.Subject |
| request\_status | String | Case.Status |
| description | String | Case.Description |
| Address | String | Case.Street\_\_c |
| Location | String | Case.City\_\_c, Case.State\_\_c |
| Zipcode | String | Case.Zip\_\_c |
| Channel | String | Case.Source |
| Image | string (URL) | Case.Customer\_Supplied\_Image\_\_c |
| cust\_first\_name | String | Contact.FirstName |
| cust\_last\_name | String | Contact.LastName |
| cust\_address | String | Contact.MailingAddress |
| cust\_zip | String | Contact.MailingPostalCode |
| cust\_phone | String | Contact.Phone |
| cust\_email | String | Contact.Email |
| request specific fields |  | Appropriate Case Field |
| Comment | String | CaseComment.CommentBody |

## Request Specific Fields and Their Mappings

|  |  |
| --- | --- |
| **Public Stuff Field** | **Mapping To Salesforce Field(Case Object)** |
| Is Light Illuminating a Parking Lot? | Is\_the\_Light\_Illuminating\_a\_Street\_\_c |
| Is the light a traffic signal? | Is\_the\_Light\_a\_Traffic\_Signal\_\_c |
| Is the light in a Park? | Is\_the\_Light\_in\_a\_Park\_\_c |
| Is the light illuminating an alley? | Is\_Light\_Illuminating\_an\_Alley\_\_c |
| Color of ~~the~~ vehicle | Color\_\_c |
| License plate # | License\_Plate\_\_c |
| Location on property | Location\_on\_Property\_\_c |
| Does it include hazardous household waste? | Trash\_Include\_Household\_Hazardous\_Waste\_\_c |
| Exact location of tree | Tree\_Location\_\_c |
| Does it include commercial hazardous waste? | Trash\_Include\_Commercial\_Hazardous\_Waste\_\_c |
| Is the trash on a vacant lot? | Is\_Trash\_on\_Vacant\_Lot\_\_c |
| Is there active construction on the site? | Is\_there\_active\_construction\_at\_the\_site\_\_c |
| What are the conditions of material? | Condition\_of\_Materials\_\_c |
| What is the type of material? | Type\_of\_Materials\_\_c |
| Description of Vehicle/Person Involved if known. | Description\_of\_Vehicle\_Involved\_in\_Dumpi\_\_c |
| Is the building occupied or vacant? | Vacant\_or\_Occupied\_\_c |
| ~~Do you hear or see running water in the hole?~~ Is there running water in the hole? | Running\_Water\_\_c |
| Is the trash on the street or the sidewalk? | Is\_Trash\_on\_Street\_or\_Sidewalk\_\_c |
| License plate state | License\_Plate\_State\_\_c |
| Condition of vehicle | General\_Condition\_\_c |
| ~~Make & Model of vehicle\*~~ Make & Model of vehicle | ~~Make~~ Model\_\_c |
| How many trees have fallen? | How\_Many\_Trees\_\_c |
| What Floor is it located on? | Floor\_\_c |
| Is it located on a rail Corridor | Rail\_Corridor\_\_c |
| **Public Stuff Field** | **Mapping To Salesforce Field(Case Object)** |
| Is it on a painted surface? | Painted\_Surface\_\_c |
| Surface Type | Surface\_Type\_\_c |
| Paint Color | Paint\_Color\_\_c |
| Property Type | Property\_Type\_\_c |
| Are you the property Owner? | Property\_Owner\_\_c |
| Is the hole in a trolley track area? | Inside\_or\_Within\_18\_of\_Trolley\_Tracks\_\_c |
| Is the light on Private Property? | Private\_Property\_\_c |
| Problem | Problem\_Type\_\_c |
| Is gas escaping from the hole? | Gas\_Escaping\_from\_the\_Hole\_\_c |
| Is the hole on an interstate highway? | On\_State\_Highway\_\_c |
| Where is the dangerous condition located on the property? | Location\_of\_Dangerous\_Condition\_\_c |
| Broken Steering Column | Is\_there\_a\_busted\_steering\_column\_\_c |
| What is the dangerous condition? | <*Not Mapped – but data is captured in Case.Description*> |
| Describe the issue | <*Not Mapped – but data is captured in Case.Description*> |
| Exact location on property | <*Not Mapped – but data is captured in Case.Description*> |
| Description of Problem | <*Not Mapped – but data is captured in Case.Description*> |
| Exact location within the park | <*Not Mapped – but data is captured in Case.Description*> |
| Name of the park | <*Not Mapped – but data is captured in Case.Description*> |
| Exact location | <*Not Mapped – but data is captured in Case.Description*> |
| Describe the hole | <*Not Mapped – but data is captured in Case.Description*> |
| Is this a bus route? | <*Not Mapped – but data is captured in Case.Description*> |
| Is the light on a wood pole? | <*Not Mapped – but data is captured in Case.Description*> |
| Is the issue in the front or rear? | <*Not Mapped – but data is captured in Case.Description*> |
| Description of exterior maintenance issues. (if applicable) | <*Not Mapped – but data is captured in Case.Description*> |
| Describe what needs to be cleaned | <*Not Mapped – but data is captured in Case.Description*> |
|  |  |

## 2.6 DEVELOPMENT

### 2.6.1 Coding Standards

All services will be developed and deployed in a packaged named ‘Philly311’.

### 2.6.2 Developers

Developers will work, via RDP, in city provided servers which will be loaded with the Software AG Integration Server and the Software AG Designer tool.

### 2.6.3 Version Control

Integration server packages will be committed to the Unisys Team Foundation Server repository, ISDEV and ISTEST environments at development & test milestones and system releases.