**TECHNICAL DESIGN DOCUMENT**

**Access to Vendor Center Role to View Invoice 06/06/2019**

**Tables of Contents:**

1. Document Information
2. Functional Overview
3. Integration involved
4. Flow Chart
5. Detailed logic flow
6. General Technical Specification
   1. Custom Fields
   2. Custom Record
   3. Saved Search
   4. Script Used
   5. Work Flow
   6. Templates
7. Script and Function Details (post development)
8. Post Deployment Activities
9. Document Change history:

1. Document Information:

|  |  |
| --- | --- |
| Document Name | **Automated Bill Payments** |
| Document Author | **Anusha Meka** |
| Document Version | 1.0 |
| Document Status | Active |
| Date Released | TBD |

1. **Functional Overview:**

Adjuster’s will find their invoices in vendor center. Link is provided to view their invoices. Two tabs are provided in the Invoice page. First tab will show all invoices and its total amount. Second tab will show their invoice breakdown.

**Flow Chart:**

*No flowchats*

1. **Integration involved:**

*No Integrations Involved*

1. **Detailed Logic Flow:**

Invoice Link is provided in saved search. The link redirects to suitlet and will view invoice field information.

The invoice is restricted to adjuster only. Adjuster will be able to find only his invoice.

**General Technical Specification:**

* 1. **Custom Fields:**

*Instructions: Provide the custom field details used if any*

|  |  |  |
| --- | --- | --- |
| **Field Name** | | **Field ID** |
|  |  | |

* 1. **Custom record:**

*Instructions: Provide the custom record details used if any*

|  |  |  |
| --- | --- | --- |
| **Record Type Name** | | **Record Type ID** |
|  |  | |

* 1. **Fee Schedule:**

|  |  |
| --- | --- |
| **Fee Schedule** | **Fee Schedule ID** |

* 1. **Item:**

|  |  |
| --- | --- |
| **Item** | **Item ID** |

* 1. **Custom Lists:**

|  |  |
| --- | --- |
| **List** | **List ID** |
|  |  |

* 1. **Saved Search:**

*No Saved searches involved*

|  |  |
| --- | --- |
| **Save Search Name** | **Save Search ID** |
| View all Bills and Link for Invoice[Do Not Delete] | customsearch479 |

* 1. **Scripts Used:**

*Instructions: Provide the scripting details used if any*

|  |  |  |
| --- | --- | --- |
| **Script Type** | **Script Name** | **Script ID** |
| Suitlet | [SUT]VendorCenterInvoiceAccess | customscript\_sut\_vendor\_center\_inv\_acces |

* 1. **Role:**

|  |  |
| --- | --- |
| **Name** | **Name ID** |
|  |  |

* 1. **Workflows Used:**

*Instructions: Provide the Workflow details used if any*

|  |  |
| --- | --- |
| **Script Name** | **Script ID** |
|  |  |

* 1. **Templates Used:**

*No templates used*

|  |  |  |  |
| --- | --- | --- | --- |
| **Template Name** | **ScriptID** | **Record** | **Purpose** |
|  |  |  |  |

## Script and Function Details (post development):

* **Script Type 1:** Suitelet
* **Record Type:**
* **Event Type:** OnRequest
* **Detailed logic flow:**

1. If context.request.method == ‘GET’ then get tranid parameter and if it is not equal to null, call function getSalesOrder as shown beow.

**try{**

**if** (context.request.method === 'GET') {

**var** tran\_id=context.request.parameters.doc\_id;

**if**(tran\_id!='' && tran\_id!=**null**){

getSalesOrder(context,tran\_id);

}**else**{

showForm(context);

}

} **else** {

showResults(context);

}

} **catch** (e) {

log.debug('onRequest\_error', 'ERROR : ' + e.message);

log.debug('ERROR', 'ERROR : ' + e);

context.response.write('<p>Invoice not billed or Claim # not found. Please check with administrator.</p>');

}

1. Get the user role of the user and if it is “Vendor Center”, create form and add all required body level and item level fields as shown below:

**var** remainingUsage = runtime.getCurrentScript().getRemainingUsage();

// log.debug('remainingUsage : ' +remainingUsage)

**var** userRole = runtime.getCurrentUser().role;

// log.debug('userRole : ' +userRole)

// log.debug('tran\_id : ' +tran\_id);

**if**(userRole == '1023' || userRole == '1009' || userRole == '1024'){

**var** so\_rec=record.load({type:'salesorder',id:tran\_id,isDynamic:**true**});

//log.debug('so\_rec : ' +so\_rec);

**var** form = ui.createForm({title : 'Claim #' + so\_rec.getValue({fieldId:'tranid'})});

///SO#

**var** so = form.addField({id : 'custpage\_tranid',type : ui.FieldType.TEXT,label : 'Claim #'});

so.defaultValue=so\_rec.getValue({fieldId:'tranid'});

so.updateDisplayType({ displayType:'DISABLED'});

///Customer

**var** cus = form.addField({id: 'custpage\_customer', label: 'Customer', source : 'customer',type : 'select'});/\*, source : 'customer'\*/

cus.defaultValue=so\_rec.getValue({fieldId:'entity'});

cus.updateDisplayType({ displayType:'DISABLED'});

**var** insured = form.addField({id: 'custpage\_adjuster', label: 'Insured Person',type : 'text'});

insured.defaultValue=so\_rec.getValue({fieldId:'custbody\_sxrd\_insured\_name'});

insured.updateDisplayType({ displayType:'DISABLED'});

**var** so\_id= form.addField({id : 'custpage\_so\_id',type : ui.FieldType.TEXT,label : 'Claim #'});

so\_id.defaultValue=tran\_id;

so\_id.updateDisplayType({ displayType:'HIDDEN'});

**var** invoiceSublist = form.addSublist({id : 'custpage\_invoice', type : ui.SublistType.INLINEEDITOR, label : 'Invoice(s)' });

invoiceSublist.addField({id : 'custpage\_invoiceid',label : 'Invoice #',type : 'TEXT'});

invoiceSublist.addField({id : 'custpage\_\_fee\_schedule',label : 'Fee Schedule',type : 'select',source:'customrecord\_sxrd\_fee\_schedule'});

invoiceSublist.addField({id : 'custpage\_invoice\_grossloss',label : 'Invoice GrossLoss',type : 'TEXT' });

invoiceSublist.addField({id : 'custpage\_amount',label : 'Amount',type : 'currency'});

**var** invoiceDetailsSublist = form.addSublist({id : 'custpage\_invoice\_details', type : ui.SublistType.INLINEEDITOR, label : 'InvoiceLineItems(s)' });

invoiceDetailsSublist.addField({id : 'custpage\_invoiceid\_details',label : 'Invoice #',type : 'TEXT'});

invoiceDetailsSublist.addField({id : 'custpage\_item',label : 'Item',type : 'select',source:'item'});

invoiceDetailsSublist.addField({id : 'custpage\_description',label : 'Description',type : 'TEXT'});

invoiceDetailsSublist.addField({id : 'custpage\_quantity',label : 'Quantity',type : 'float' });

invoiceDetailsSublist.addField({id : 'custpage\_amount',label : 'Amount',type : 'currency'});

invoiceDetailsSublist.addField({id : 'custpage\_taxamount',label : 'Tax Amount',type : 'currency'});

invoiceDetailsSublist.addField({id : 'custpage\_adjuster\_amount',label : 'Adjuster Amount',type : 'currency'});

1. Get the line count using

**var** numLinesMainTab = invoiceSublist.lineCount;

1. Now search for the claim and get invoice body fields and set them in the custom fields. Now search for the invoice and set those fields in the custom item fields.
2. Add new button to search new invoice.
3. The “showResults” function will get claim number and searches for the claim. If claim number is entered incorrect, message will be shown as below:

context.response.write('<p>Claim Number not found. Please check your claimnumber or contact administrator.</p>');

}

}**else**{

context.response.write('<p>There is no match </p>');

Post Deployment Activities:

*No post deployment activities*

|  |  |
| --- | --- |
| **Script Name** | **Activity** |
|  |  |
|  |  |

1. **Document Change history:**

|  |  |  |  |
| --- | --- | --- | --- |
| Author | **Version** | **Date** | **Description of Changes** |
| Anusha Meka | 1.0 | 01/03/2018 | Document Created |
|  |  |  |  |
|  |  |  |  |