**TECHNICAL DESIGN DOCUMENT**

**Send PDF through an email when adjuster assigned to NatGen claims**

**03/22/2019**

**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:**

Send an email to NatGen attaching PDF when a claim is assigned to an adjuster or reassigned to adjuater

**Flow Chart:**

*No flowchats*

1. **Integration involved:**

*No Integrations Involved*

1. **Detailed Logic Flow:**

When an email when a claim is created attaching PDF.

When a claim is reassigned to new adjuster, send an email attaching PDF.

**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** |
|  |  |

* 1. **Scripts Used:**

*Instructions: Provide the scripting details used if any*

|  |  |  |
| --- | --- | --- |
| **Script Type** | **Script Name** | **Script ID** |
| User Event | [UES]claimToPDF2\_0 | customscript\_ues\_claim\_to\_pdf |

* 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** |
| Custom Sales Order PDF/HTML Template | CUSTTMPL\_100\_4667405\_560 | Claim |  |

## Script and Function Details (post development):

* **Script Type 1:** UserEvent
* **Record Type:** Claim (Sales Order)
* **Event Type:** Before Submit, After Submit
* **Detailed logic flow:**

1. Check the value of the context if it is CRETE or EDIT.
2. Get the value of the record id and customer (entity) using

if(context.type == context.UserEventType.CREATE || context.type == context.UserEventType.EDIT){

var object = context.newRecord;

var salesorderInternalid = context.newRecord.id;

var documentnumber = object.getValue({

fieldId: 'tranid'

});

var customer = object.getValue({

fieldId: 'entity'

1. Now get the selected adjuster email id and phone number using vendor lookup and set them to custom fields present in claim record.

var adjusterPhone = search.lookupFields({ // Search for Invoice Number

type: search.Type.VENDOR,

id: adjuster,

columns: 'phone'

});

// log.debug({title: 'adjusterPhone',details: adjusterPhone});

var adjusterEmail = search.lookupFields({ // Search for Invoice Number

type: search.Type.VENDOR,

id: adjuster,

columns: 'email'

});

// log.debug({title: 'adjusterPhone',details: adjusterPhone});

var email = adjusterEmail.email;

// log.debug({title: 'adjusterEmail',details: email});

var phone = adjusterPhone.phone

// log.debug({title: 'adjusterPhone',details: phone});

object.setValue({fieldId:'custbody\_adjuster\_email',value:email});

object.setValue({fieldId:'custbody\_adjuster\_phone',value:phone});

1. If the context is CREATE and if the adjuster not equal to null, then send email attaching PDF to it.

if(context.type == context.UserEventType.CREATE){

if(adjuster != "" || adjuster != "null" || adjuster != 44154){

//Print --> convert the record into a PDF or HTML object

var TransactionPdfObj = render.transaction({

entityId:salesorderInternalid,

printMode:render.PrintMode.PDF

});

//Email PDF object to recipient

email.send({

author: 22085,

recipients: 'claims@ngic.com',

subject:'Claim: '+documentnumber+'-NGLS-IA Adjuster Acknowledgement',

body:'Claim Number: '+documentnumber+'\n\n IA Adjuster acknowledgement attached to this message.\n\n\n Thank you,\n\nFourseventy Claims',

replyTo: 'no-reply@470.com',

attachments:[TransactionPdfObj]

});

}

1. If context mode is EDIT, then search the systemnotes on claim and find out the reassigned adjuster. And send an email attaching PDF to it.

var salesorderSearchObj = search.create({

type: "salesorder",

filters:

[

["type","anyof","SalesOrd"],

"AND",

["mainline","is","T"],

"AND",

["systemnotes.field","anyof","CUSTBODY\_SXRD\_ADJUSTER"],

"AND",

["name","anyof","214","47"],

"AND",

["formulatext: {systemnotes.type}","is","change"],

"AND",

["internalidnumber","equalto",salesorderInternalid]

],

columns:

[

search.createColumn({

name: "trandate",

sort: search.Sort.ASC,

label: "Date"

}),

search.createColumn({name: "datecreated", label: "Date Created"}),

search.createColumn({name: "tranid", label: "Document Number"}),

search.createColumn({name: "type", label: "Type"}),

search.createColumn({

name: "type",

join: "systemNotes",

label: "Type"

}),

search.createColumn({

name: "field",

join: "systemNotes",

label: "Field"

}),

search.createColumn({

name: "oldvalue",

join: "systemNotes",

label: "Old Value"

}),

search.createColumn({

name: "newvalue",

join: "systemNotes",

label: "New Value"

}),

search.createColumn({

name: "context",

join: "systemNotes",

label: "Context"

}),

search.createColumn({name: "custbody\_sxrd\_adjuster", label: "Adjuster"})

]

});

var resultset = salesorderSearchObj.run();

/\* log.debug({

title: 'resultset',

details: resultset

});\*/

var results = resultset.getRange(0, 999);

if (results) {

/\* log.debug({

title: 'results',

details: results

});\*/

for (var i = 0; results != null && i < results.length; i++){

/\* log.debug({

title: 'i',

details: i

});\*/

var result = results[i];

/\* log.debug({

title: 'result',

details: result

});\*/

var internalid1 = results[i].getValue('internalid');

// log.debug({title: 'internalid1',details: internalid1});

if(adjuster != "" || adjuster != "null"){

//Print --> convert the record into a PDF or HTML object

var TransactionPdfObj = render.transaction({

entityId:salesorderInternalid,

printMode:render.PrintMode.PDF

});

//Email PDF object to recipient

email.send({

author:22085,

recipients: 'claims@ngic.com',

subject:'Claim: '+documentnumber+'-NGLS-IA Adjuster Acknowledgement',

body:'Claim Number: '+documentnumber+'\n\n IA Adjuster acknowledgement attached to this message.\n\n\n Thank you,\n\nFourseventy Claims',

replyTo: 'no-reply@470.com',

attachments:[TransactionPdfObj]

});

break;

## 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 |
|  |  |  |  |
|  |  |  |  |