##### Audit and Assurance Questionnaire

**Description:** When an opportunity or engagement has an Audit and Assurance related product, a set of additional fields/questionnaire should be visible for the user. In order to accomplish this, we have 2 objects that serve to store the configuration of the questionnaire and its values, **Audit and Assurance Questions** and **Audit and Assurance Questions Template** also the objects **Audit and Assurance Section** to indicate in which section we will display them.

Per each record on the *Audit and Assurance Questions Template* object we define the specific product where this question applies, the object type (Opportunity, Engagement), the order, type of question (text, picklist, multi-picklist, lookup), etc.

Once an opportunity/engagement is saved with a product found in the Audit and Assurance Questions Template records with its field *Type* as *Audit And Assurance*, all those are copied into records of object type *Audit and Assurance Questions* records with its field *Type* as *Audit And Assurance*. These questions are then displayed in the page layout, in the A&A Questions tab.

On the questions objects, there is a flag that indicates if the field is mandatory for approval of engagement partner and another field that indicates if it is mandatory for conflict check. When a user clicks on save on the Audit and Assurance Questionnaire, all the mandatory fields are validated and if all of them are filled, the *A&A Approval Partner Completed* and *A&A Conflict Check Completed* fields in Opportunity or Engagement are set as true.

If the product of the opportunity/engagement is changed, the questions are generated again if is also an Audit and Assurance product, if not, they are just deleted and a message that no audit product is related appears in place for the questions.

When cloning an opportunity/engagement, all these questions are also cloned to the new record with all of its values. On the other hand, clustering engagement set the primary product as empty, so the new record would not have any Audit question related to it.

All the Audit and Assurance Products and the questions related to them are defined on the tabs in the excel file attached to the user stories.

**User Stories**: From US-5095 to US-5104 for opportunity questions. Then from US-5105 to US-5114 for engagement questions, with the US-5103 and US-5113 canceled

**Triggers:** Saving/updating an Opportunity/Engagement with an Audit and Assurance product set in its Product/Primary product field, that is also found on the Product field of the Audit and Assurance Questions Template records.

Figure 4. Account Technical Flow

**Metadata**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Audit and Assurance Questions Template**(DEAL\_AuditAndAssuranceQuestionsTemplate\_\_c):   * DEAL\_AARequireForConflict\_\_c * DEAL\_AARequireForPartner\_\_c * DEAL\_AudiAndAssureanceSection\_\_c * DEAL\_AudiAndAssureanceSectionGerman\_\_c * DEAL\_Detail\_\_c * DEAL\_DetailGerman\_\_c * DEAL\_ErrorMessage\_\_c * DEAL\_ErrorMessageGerman\_\_c * DEAL\_HelpText\_\_c * DEAL\_InputType\_\_c * DEAL\_InputValueGerman\_\_c * DEAL\_InputValues\_\_c * DEAL\_isRequired\_\_c * DEAL\_ObjectType\_\_c * DEAL\_Product\_\_c * DEAL\_QuestionOrderNumber\_\_c * DEAL\_ReadOnly\_\_c * DEAL\_RegEx\_Validation\_\_c * DEAL\_Type\_\_c   **Audit and Assurance Question**(DEAL\_AuditAndAssuranceQuestions\_\_c):   * DEAL\_AARequireForConflict\_\_c * DEAL\_AARequireForPartner\_\_c * DEAL\_AudiAndAssureanceSection\_\_c * DEAL\_AudiAndAssureanceSectionGerman\_\_c * DEAL\_Detail\_\_c * DEAL\_DetailGerman\_\_c * DEAL\_Engagement\_\_c * DEAL\_ErrorMessage\_\_c * DEAL\_ErrorMessageGerman\_\_c * DEAL\_HelpText\_\_c * DEAL\_InputType\_\_c * DEAL\_InputValueGerman\_\_c * DEAL\_InputValues\_\_c * DEAL\_isRequired\_\_c * DEAL\_Opportunity\_\_c * DEAL\_ObjectType\_\_c * DEAL\_Product\_\_c * DEAL\_QuestionOrderNumber\_\_c * DEAL\_ReadOnly\_\_c * DEAL\_RegEx\_Validation\_\_c * DEAL\_Type\_\_c   **Audit and Assurance Section**(DEAL\_AuditAssuranceSection\_\_c):  DEAL\_HelpText\_\_c  DEAL\_Language\_\_c  DEAL\_SectionName\_\_c  DEAL\_SectionOrderNumber\_\_c  **Opportunity**:  DEAL\_AAApprovalPartnerCompleted\_\_c  DEAL\_AAConflictCheckCompleted\_\_c  DEAL\_ServiceLineProduct\_\_c  **Engagement**(DEAL\_Engagement\_\_c):  DEAL\_AAApprovalPartnerCompleted\_\_c  DEAL\_AAConflictCheckCompleted\_\_c  DEAL\_PrimaryProduct\_\_c |
| Lightning Component | DEAL\_AuditAndAssurance  DEAL\_AuditAndAssuranceValues  DEAL\_Lookup |
| Apex class | DEAL\_AuditAndAssuranceController  DEAL\_OpportunityTriggerHandler methods:   * additionalAuditAssuranceData * additionalAuditAssuranceDataUpdate   DEAL\_EngagementTriggerHandler methods:   * additionalAuditAssuranceData * additionalAuditAssuranceDataUpdate   DEAL\_DynamicLookup |
| Apex Trigger | DEAL\_OpportunityTrigger  DEAL\_EngagementTrigger |

##### Risk Assesment and Risk Management Questionnaire(11100 Block)

**Description:** Works almost exactly the same as the Audit and Assurance Questionnaire, only with the differences explained below:

When creating the *Audit and Assurance Questions Template* records, the questions should be created with the ***Type*** *Risk Assessment* or with *Risk Management*.

Audit and Assurance questions only are within one section, but this questions of type Risk Assesment can have sections and subsections when displaying on screen.

The input from user is saved as soon as it leaves the focus of the field.

Questions, depending on it’s type, are displayed within the Opportunity and Engagement record pages in the tabs *Risk Assesment* and *Risk Management*. Each tab has its own set of required fields that are validated when saving to indicate on the Opportunity/Engagement on the fields *Risk Assessment Completed* and *Risk Management Completed*.

All the products and the questions related to them are defined on the tabs in the excel file attached to the user stories.

**User Stories**: User Stories from US-5168 to US-5171

**Triggers:** Saving/updating an Opportunity/Engagement with a Risk Assesment/Management product set in its Product/Primary product field, that is also found on the Product field of the Audit and Assurance Questions Template records of Type Risk Assesment/Management.

**Metadata**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Audit and Assurance Questions Template**(DEAL\_AuditAndAssuranceQuestionsTemplate\_\_c):   * DEAL\_AARequireForConflict\_\_c * DEAL\_AARequireForPartner\_\_c * DEAL\_AudiAndAssureanceSection\_\_c * DEAL\_AudiAndAssureanceSectionGerman\_\_c * DEAL\_Detail\_\_c * DEAL\_DetailGerman\_\_c * DEAL\_ErrorMessage\_\_c * DEAL\_ErrorMessageGerman\_\_c * DEAL\_HelpText\_\_c * DEAL\_InputType\_\_c * DEAL\_InputValueGerman\_\_c * DEAL\_InputValues\_\_c * DEAL\_isRequired\_\_c * DEAL\_ObjectType\_\_c * DEAL\_Product\_\_c * DEAL\_QuestionOrderNumber\_\_c * DEAL\_ReadOnly\_\_c * DEAL\_RegEx\_Validation\_\_c * DEAL\_Type\_\_c   **Audit and Assurance Question**(DEAL\_AuditAndAssuranceQuestions\_\_c):   * DEAL\_AARequireForConflict\_\_c * DEAL\_AARequireForPartner\_\_c * DEAL\_AudiAndAssureanceSection\_\_c * DEAL\_AudiAndAssureanceSectionGerman\_\_c * DEAL\_Detail\_\_c * DEAL\_DetailGerman\_\_c * DEAL\_Engagement\_\_c * DEAL\_ErrorMessage\_\_c * DEAL\_ErrorMessageGerman\_\_c * DEAL\_HelpText\_\_c * DEAL\_InputType\_\_c * DEAL\_InputValueGerman\_\_c * DEAL\_InputValues\_\_c * DEAL\_isRequired\_\_c * DEAL\_Opportunity\_\_c * DEAL\_ObjectType\_\_c * DEAL\_Product\_\_c * DEAL\_QuestionOrderNumber\_\_c * DEAL\_ReadOnly\_\_c * DEAL\_RegEx\_Validation\_\_c * DEAL\_Type\_\_c   **Audit and Assurance Section**(DEAL\_AuditAssuranceSection\_\_c):  DEAL\_HelpText\_\_c  DEAL\_Language\_\_c  DEAL\_SectionName\_\_c  DEAL\_SectionOrderNumber\_\_c  DEAL\_SectionParent\_\_c  DEAL\_Subsection\_\_c  **Opportunity**:  DEAL\_AAApprovalPartnerCompleted\_\_c  DEAL\_AAConflictCheckCompleted\_\_c  DEAL\_ServiceLineProduct\_\_c  **Engagement**(DEAL\_Engagement\_\_c):  DEAL\_AAApprovalPartnerCompleted\_\_c  DEAL\_AAConflictCheckCompleted\_\_c  DEAL\_PrimaryProduct\_\_c |
| Lightning Component | DEAL\_AuditManagementAndAssesmentFields  DEAL\_AuditManagementAndAssesmentFieldValues  DEAL\_Lookup |
| Apex class | DEAL\_AudManageAndAssesController  DEAL\_OpportunityTriggerHandler methods:   * additionalAuditAssuranceData * additionalAuditAssuranceDataUpdate   DEAL\_EngagementTriggerHandler methods:   * additionalAuditAssuranceData * additionalAuditAssuranceDataUpdate   DEAL\_DynamicLookup |
| Apex Trigger | DEAL\_OpportunityTrigger  DEAL\_EngagementTrigger |

##### Change Responsibility Tool

**Description:** An admin is able to search object responsibilities so he can change them. Users responsibles should be searched by user name, sure name, related office, email, role/responsibility and object type. An specific record or multiple records can be selected to change the responsibility to other user. These are the current responsibilities by object:

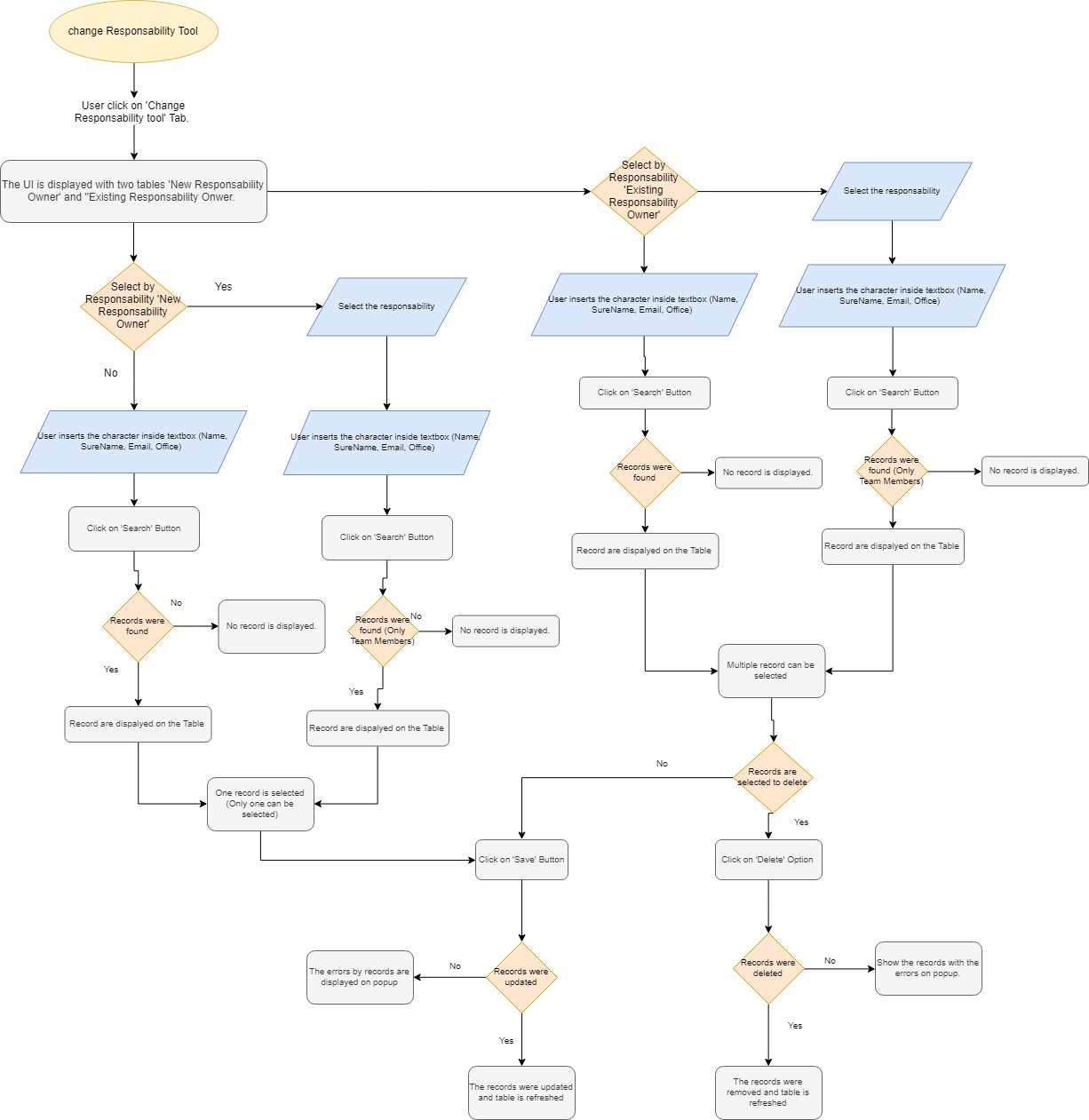
* **Account:** Account Admin, Account Manager, Lead Audit & Assurance Partner, Business Developer 1, Business Developer 2, Lead Rick Advisory Partner, Lead Financial Advisory Partner, Lead Consulting Partner, Account Owner, Lead Relationship Partner, Lead Tax & Legal Partner, Lead Client Service Partner.
* **Campaign:** Asset Developer, Campaign Owner from Business 1, Annual Plan Owner 2, Sub-Campaign Owner from Business 1, Sub-Campaign Owner from Business 2, Business Campaign Responsible, Business Development Plan Owner, Asset Second Owner, Business Sub-Campaign Owner, Campaign Owner, Business Development Plan Owner 2, Business Sub-Campaign Responsible, Sub-Campaign Marketing Owner, Campaign Marketing Owner, Business Campaign Owner, Asset Marketing Owner, Annual Plan Owner 1, Campaign Owner from Business 2.
* **Contact:** Contact Owner.
* **Business Relationship:** Relationship Admin, Relationship Partner, Relationship Owner.
* **Target Group:** Target Group Owner 1, Target Group Owner 2, Target Group Owner 3.
* **Team Member:** Engagement Partner, Engagement Manager, Engagement Admin, Sales Partner, Sales Manager, Opportunity Partner, Opportunity Manager, Opportunity Admin, Lead Partner, Lead Manager, Lead Admin, Pursuit, Relationship Supporter, Signatory of the left, Signatory of the right, Engagement Owner, Co-Manager, Co-Partner, LCSP, Account Owner, Account Manager, Account Admin, Business Developer, Lead Audit & Assurance Partner, Lead Tax & Legal Partner, Lead Financial Advisory Partner, Lead Risk Advisory Partner, Lead Consulting Partner, Lead Relationship Partner, Group Account Admin, Second Reviewer Partner (Tax), EQCR (RA), QA Partner (Consulting), Concurrent Reviewer (FA), Special Risk Reviewer (FA), Lead Member Firm, Member Firm.
* **Task:** Task Owner.

**User Stories: US-4732, US-5476, US-5589.**

**Trigger:** Update and remove the related user on the selected responsibility to be replaced by another user.

**Metadata:**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Account** (DEAL\_AccountAdmin\_\_c, DEAL\_AccountManager\_\_c, DEAL\_LeadAuditAssurancePartner\_\_c, DEAL\_BusinessDeveloper2\_\_c, DEAL\_LeadRiskAdvisoryPartner\_\_c, DEAL\_LeadFinancialAdvisoryPartner\_\_c, DEAL\_LeadConsultingPartner\_\_c, OwnerId, DEAL\_LeadRelationshipPartner\_\_c, DEAL\_LeadTaxLegalPartner\_\_c, DEAL\_BusinessDeveloper1\_\_c, DEAL\_LeadClientServicePartner\_\_c). **Campaign**(DEAL\_mkt\_AssetDeveloper\_\_c, DEAL\_mkt\_CampaignOwnerFromBusiness1\_\_c, DEAL\_mkt\_AnnualPlanOwner2\_\_c, DEAL\_mkt\_SubCampaignOwnerBusiness2\_\_c, DEAL\_mkt\_SubCampaignOwnerBusiness1\_\_c, DEAL\_mkt\_BusisCampaignResponsible\_\_c, DEAL\_mkt\_BDPOwner\_\_c, DEAL\_mkt\_AssetSecondOwner\_\_c, DEAL\_mkt\_BussSubCampaignOwner\_\_c, DEAL\_mkt\_BDPOwner2\_\_c, DEAL\_mkt\_BusinessSubCampaignResponsible\_\_c, DEAL\_mkt\_SubCampaignMarketingOwner\_\_c, DEAL\_mkt\_CampaignMarketingOwner\_\_c, DEAL\_mkt\_BussCampaignOwner\_\_c, DEAL\_mkt\_AssetMarketingOwner\_\_c, DEAL\_mkt\_AnnualPlanOwner1\_\_c, DEAL\_mkt\_CampaignOwnerFromBusiness2\_\_c). **Contact** (OwnerId). **DEAL\_BusinessRelationship\_\_c** (DEAL\_BusinessRelationshipAdmin\_\_c, DEAL\_BusinessRelationshipPartner\_\_c, OwnerId). **DEAL\_mkt\_TargetGroup\_\_c** (DEAL\_mkt\_TargetGroupOwner1\_\_c, DEAL\_mkt\_TargetGroupOwner2\_\_c, DEAL\_mkt\_TargetGroupOwner3\_\_c). **DEAL\_Team\_\_c** (DEAL\_TeamMemberUser\_\_c). **Task** (OwnerId). |
| Apex Class | DEAL\_ObjectResponsabilityController (All Methods) |
| Lightning component | DEAL\_ObjectResponsability |
| Custom Metadata | DEAL\_Responsability\_Query |



##### Business Developer Tool

**Description:** Each global ultimate parent account or single company on the org has two user lookup named business developer 1 and 2. This component lets the user set them, edit them or delete them massively. Only the Client Program Global accounts are searched, as indicated in the US acceptance criteria.

Right now, to maintain these business developers, they are stored in an object named *Business Developer*. A Batch process that should run every hour, search for changes on these records to set/update/delete the accounts that match. The business developer record creations works as following:

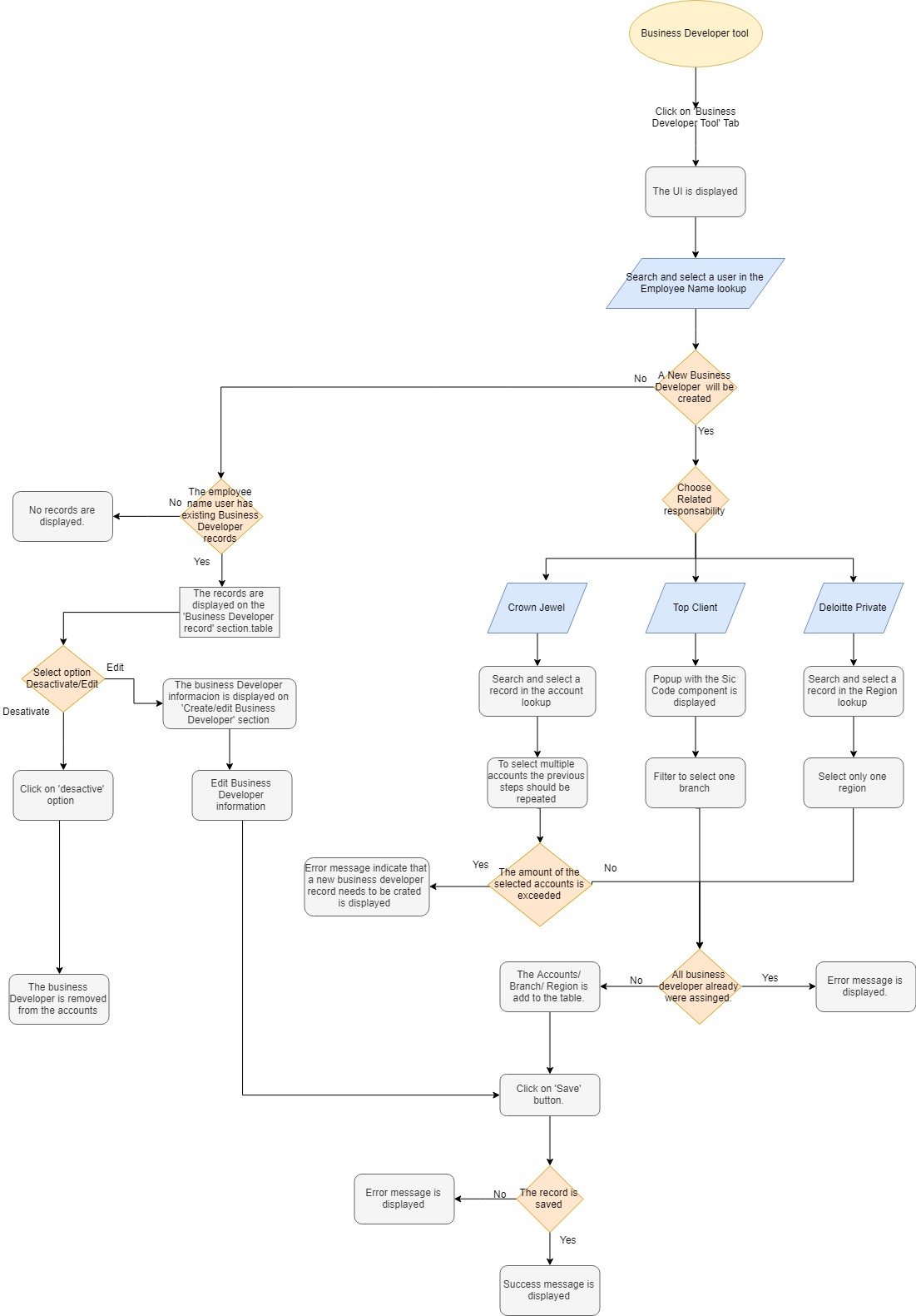
To create a Business developer record in the component, a user should be selected in the *Employee Name* lookup, once selected, the section *Create/Edit Business Developper* should be opened. Then, a related responsibility must be selected and works as explained:

* Crown Jewel: When selecting this a lookup named *Account* is displayed. Whenever an account is selected from here, is added to the table displayed bellow the other fields. After saving, all the account ids are stored in the *Records* field (Deal\_Records\_\_c). Once the Batch process runs, those accounts stored in the *Records* field and whose *Client Program Global* is *Crown Jewel,* will have its *Business Developer* 1 or 2 filled with the *Employee Name* user selected.
* Top Client: When selecting this the *SIC Code* component is displayed to let the user select a Branch. After saving, the Branch id is stored in the *Branch* field (Deal\_Branch\_\_c). Once the Batch process runs, those accounts whose *Client Program Global* is *Top Client* and its Branch is the one selected*,* will have its *Business Developer* 1 or 2 filled with the *Employee Name* user selected.
* Deloitte Private: When selecting a lookup named *Region* is displayed. After saving, the Region id is stored in the *Region*  field (Deal\_Region\_\_c). Once the Batch process runs, those accounts whose *Client Program Global* is *Deloitte Private* and its Region is the one selected*,* will have its *Business Developer* 1 or 2 filled with the *Employee Name* user selected.

When saving the Business Developer record, an account is searched that matches the business developer criteria and is taken as a sample account to look if it has any Business Developer lookup empty (all accounts that match the criteria should have the same values on those lookups). If no Business Developer lookup is empty, the system should show a message asking the user to edit an existing business developer or to delete it.

To edit or delete business developer records, after selecting a user in the *Employee Name* lookup, it will display all the existing records in the ‘*Business Developer Records*’ section. After selecting the desire record, it will be available to edit in the *Create/Edit* section.

**User story:** US-4839.

**Triggers:** A batch process that runs every hour and search for changes on Business Developer records.

**Medadata:**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Account** (fields: DEAL\_BusinessDeveloper1 \_\_c, DEAL\_BusinesDeveloper2\_\_c). **DEAL\_Branch\_\_c** (Id). **DEAL\_Region\_\_c** (Id).  **DEAL\_BusinessDeveloper\_\_c**  Fields:  DEAL\_Active\_\_c  DEAL\_Branch\_\_c  DEAL\_EmployeeName\_\_c  DEAL\_Note\_\_c  DEAL\_Records\_\_c  DEAL\_Region\_\_c  DEAL\_RelatedResponsibility\_\_c |
| Lightning component | DEAL\_BusinessDeveloperTool\_\_c, DEAL\_EmployeeNameEvent, DEAL\_BusinessDeveloperRecords, DEAL\_dynamicLookup. |
| Apex class | DEAL\_BusinessDeveloperController, DEAL\_DynamicLookup. |

##### Cluster Engagement

**Description:** Cluster an engagement to let the user split a bigger engagement into more specific ones.

To allow an Engagement to be clustered, it must be granted explicit permission to do so by the engagement partner or manager. The user who granted this permission and the hour of this action are then saved on the engagement record. After this, when a user wants to cluster the Engagement, a button on the Engagement Hierarchy component at the right side, named Cluster, is displayed.

This button starts process of clustering and validates that all the criteria conditions are met, then the related lists specified in the custom metadata are cloned to the new Engagement. Also a deifned set of fields are set as null, like the primary product. Setting those values as null is one of the main differences with the cloning process.

**User Stories**: US-4933, US-4514

**Triggers:** Pressing the *Cluster* button; automatically creates the new record.

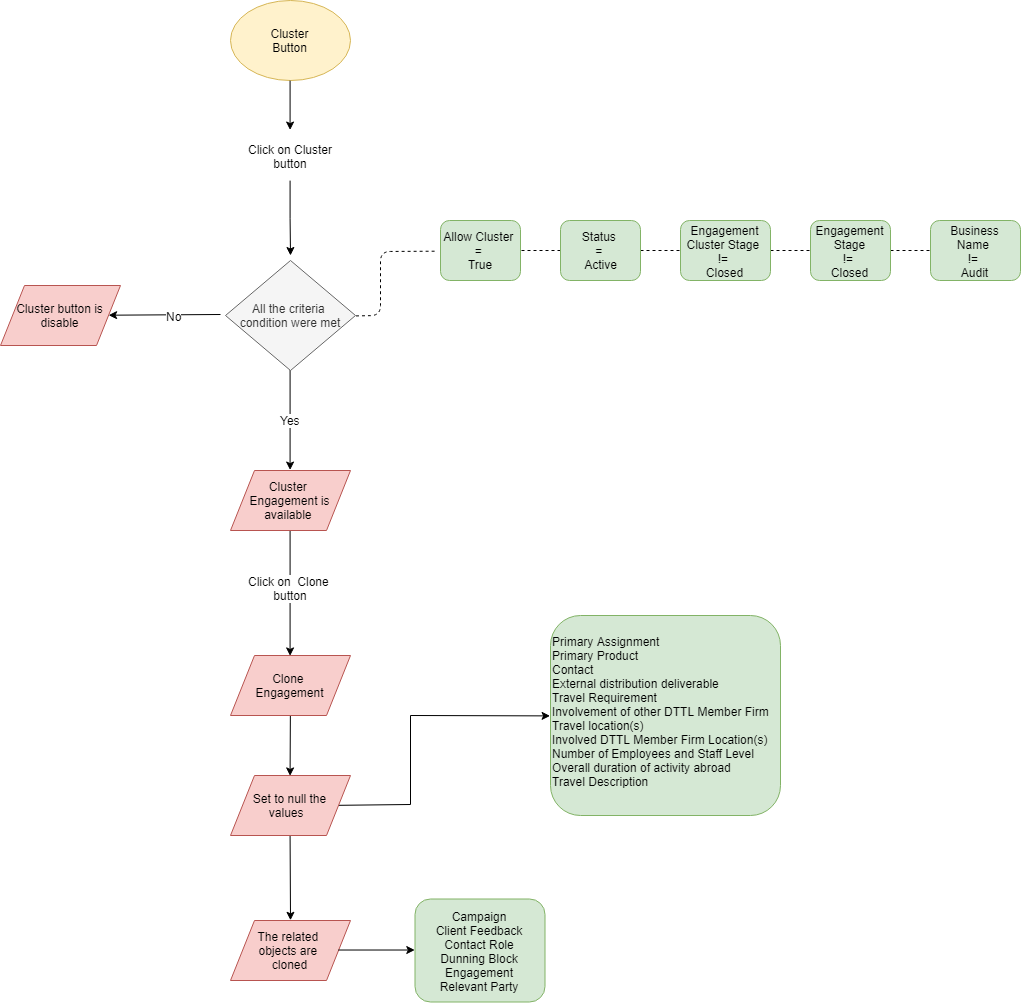


Figure 4. Cluster Technical Flow

**Metadata**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Engagement** (DEAL\_AllowCluster\_\_c , DEAL\_AllowClusterDate\_\_c ,DEAL\_PrimaryAssignment\_\_c, DEAL\_AllowClusterEmployee\_\_c, DEAL\_Status\_\_c, DEAL\_EngagementStage\_\_c, DEAL\_PrimaryProduct\_\_c, DEAL\_Contact\_\_c, DEAL\_ExternalDistributionDeliverable\_\_c, DEAL\_DeliverableDistributionLocation\_\_c, DEAL\_TravelRequirement\_\_c, DEAL\_InvolvementOfOtherDTTLMemberFirm\_\_c, DEAL\_TravelLocations\_\_c, DEAL\_InvolvedDTTLMemberFirmLocations\_\_c, DEAL\_NumberOfEmployeesAndStaffLevel\_\_c, DEAL\_EngagementCluster\_\_c, DEAL\_ParentId\_\_c, DEAL\_OverallDurationOfActivityAbroad\_\_c, DEAL\_TravelDescription\_\_c) |
| Lightning Component | DEAL\_EngagementHierarchy.cmp |
| Apex Class | DEAL\_EngagementHierarchyController (methods: getEngagementParent, getEngagementBusiness, doCloneEngagement, doClone, cloneChildren, prepareSubqueries) |
| Custom Setting | DEAL\_AuditProducts |
| Custom Metadata | Clone\_Engagements\_\_mdt(Campaign, DEAL\_ClientFeedback\_c, DEAL\_ContactRole\_c, DEAL\_DunningBlock\_c, DEAL\_Engagement\_c, DEAL\_RelevantParty\_c) |

##### Private Equity Investor and Private Equity Portfolios

**Description:**  Identifies an account as Private Equity Investor and list to the user all its related PE Portfolios. To do so, a branch with a sector with ‘Private Equity’ on its name should be selected in the desired account to make it PE Investor. There is a formula field (*Private Equity Investor*) that identifies if an account is PE Investor based on its sic code sector name.

Once an account is made PE Investor, on its PE Portfolio related at the right side, the PE Portfolio records can be created, selecting an existing Global Ultimate Parent or single account (Single accounts are those without parent account and without subsidiaries) as the PE Portfolio. There is no limit for the PE Portfolios that can be created. As soon as the PE Portfolio is created, this PE Portfolio account is flagged as PE Portfolio with a checkbox (*PE Portfolio* field). This value must flow through all the subsidiaries of the PE Portfolio account.

Also when saving the record, the LCSP of the PE Investor account and the mail of said LCSP are copied in to the PE Portfolio account.

On account record pages, there’s a component in the *More* tab within the *PE Investor Hierarchies* that validate if account is PE Investor account, if it is, displays a button that opens a Salesforce Report with the PE Investor account and all its related PE Portfolios and subsidiaries of those. The button its on a component due to the inability of Quick Actions to allow redirections to an specific URL with custom parameters, cause the button opens the report link adding the parameter to filter per the current account.

Whenever a new engagement or opportunity is created for the PE Portfolio accounts or its subsidiaries, an email must be sent to the LCSP of the PE Investor. This is achieved on a process builder that looks on the field *LCSP of Private Investor* that is saved when the PE Portfolio is created as described before. If the account is subsidiary of a PE Portfolio, the process builder looks at the LCSP of PE Investor of its ultimate parent.

**User Stories**: US-4790, US-4833, US-5319, US-4798, US-4830, US-4831

**Triggers:** Selecting a SIC Code with a ‘Private Equity’ sector within an account makes it a ‘Private Equity Investor’ as per business process, no record types involved. When creating PE Portfolio records for a PE Investor, they’re flagged as PE Portfolios and get the LCSP of the PE Investor copied to them. When creating a new Opportunity or Engagement for a PE Portfolio or any of its subsidiaries, a mail is sent to the LCSP of PE Investor. When an account is identified as PE Investor a button appears on the *More* tab on the record page to take the user to a Salesforce Report where the PE Investor, its related PE Portfolios and their account hierarchies are shown.

**Medadata:**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Account**  DEAL\_PrivateEquityInvestor\_\_c  DEAL\_PEPortfolio\_\_c  DEAL\_PEPortfolioCount\_\_c  DEAL\_LCSPPrivateInvestor\_\_c  DEAL\_FullNameLCSPPrivateInvestor\_\_c  **Opportunity**  DEAL\_PrivateEquityInvestorEmail\_\_c  **Engament**  DEAL\_PrivateEquityInvestorEmail\_\_c  **PE Portfolio**  DEAL\_Active\_\_c  DEAL\_Date\_\_c  DEAL\_LCSPInvestor\_\_c  DEAL\_PEPAccount\_\_c  DEAL\_PEPortfolioA\_\_c |
| Email Alert | DEAL\_EmailAlertEngPEPortfolio  DEAL\_PEPortafolioEngagement  DEAL\_PEPortafolioEngDE  DEAL\_EmailAlertOppPEPortfolio  DEAL\_EmailAlertOppPEPortfolioDE |
| Email Template | DEAL\_PEPortafolioEngagement  DEAL\_PEPortafolioOpoortunity  DEAL\_PEPortafolioOpoortunityDE |
| Process Builder | DEAL LSCP Investor  DEAL Update Private Equity Investor Email Engagement DE  DEAL Update Private Equity Investor Email Engagement ENG  DEAL Update Private Equity Investor Email Opp DE  DEAL Update Private Equity Investor Email Opp Eng  Private Investor LCSP |

##### Sharepoint Interface

**Description:**  A sharepoint iframe was added to the page layouts of Leads, Opportunities, Engagements, Accounts and Contacts on the tab *Document Manager* to let the users upload/download or read documents on records. This iframe is in a Lightning component where we are getting info from the record and sending it to the Sharepoint Site. These are the paremeters required by Sharepoint and the values we are sending:

* customerID: Account number on account records, on the rest of objects the related account number
* companyID: Account number on account records, on the rest of objects the related account number
* dealID: Record id
* groupID: Account number of the Global Ultimate Parent on account records, on the rest of objects the Global Ultimate Parent of the related account
* title: Name of the record
* link: Salesforce link to the current record
* activity: Type of object of the current record
* AccountName: Account name on account records, on the rest of objects the related account name

These parameters are gathered in a single javascript object as fields, this object is then encoded as JSON, after this is URL encoded and to finish is Base64 encoded before sending it.

This is the link we received to sharepoint and the one is currently in use:

<https://portal-qa.de.deloitte.com/sites/deal-customer-documents/SitePages/siteredirect.aspx?mashup=1>

Later in time, after the first implementation of this, it was required to make a demo to not only to sent information to Sharepoint, but also to get information from the iframe in return. For this a Visualforce page was created, cause there was the need to add a window even listener to get the iframe messages and this listener feature is blocked in lightning components.

As per the current status, this test of comunication between systems is achieved, with only the handling of the permission assignment that should be sent to sharepoint and the encription of the salesforce side pending.

For the afore mentioned requirement of comunication, the following information was provided by Christian Loebbert, who is the one that works on the Sharepoint site, to send new parameters:

***Communication:***

1. *Build the parameter object similar to the known Document Manager integration  
   - parameters.customerID  
   - parameters.AccountName  
   - parameters.dealID  
   - parameters.groupID  
   - parameters.companyID  
   - parameters.activity  
   - parameters.contactID  
   - parameters.title  
   - parameters.link  
   - parameters.action  
   - parameters.confidential  
   - parameters.username  
   - parameters.create\_folder*

*The “action”, “username”, “create\_folder” and “confidential” parameters are the new ones.*

*Parameter “****confidential****” remove all default permissions e.g Owners, Members, Visitors from the folder if the folder is inheriting the permissions from the account site. Otherwise – if custom permissions are already set – this parameter will be ignored.*

*Parameter “****action****” will be ignored in Document Manager and can have the following values (if necessary, we ca define more):*

* *add\_read*
  + *Add Read permission level to the user (parameter username is required)*
* *add\_contribute*
  + *Add Contribute (Read/Write) permission level to the user and deny deletion or renaming of files (parameter username is required)*
* *add\_fullcontrol*
  + *Add FullControl permission level to the user. This user can do everything with the folder incl. delete files and manage permissions outside SalesForce (can be discussed; parameter username is required)*
* *remove\_permission*
  + *Remove all permission from the user on the specific folder (parameter username is required)*
* *reset\_permissions*
  + *Remove all custom permissions and inherit the default permissions from site.*
* *init*
  + *Test the interface connection. When called, it call the EventListener in SalesForce an send an object with code,text and the values 0,”Ready”.*

*Parameter “****username****” is the corresponding Deloitte user e.g. cloebbert*

*Parameter “****create\_folder****” creates the Document Manager when setting permissions. That means the customer site and the activity folder will be created and then the “action” will be applied.*

**User Stories**: US-4786, US-4455

**Triggers:** Clicking on the *Document Manager* tab on a Lead, Opportunity, Engagement, Account or Contact record.

**Medadata:**

|  |  |
| --- | --- |
| Type | Name |
| Objects and Fields | **Account**:   * Id * Name * DEAL\_AccountNumber\_\_c * DEAL\_GUPNumber\_\_c   **Lead:**   * Id * Name * DEAL\_Account\_\_r.Name * DEAL\_Account\_\_r.DEAL\_AccountNumber\_\_c * DEAL\_Account\_\_r.DEAL\_GUPNumber\_\_c   **Opportunity:**   * Id * Name * Account.Name * Account.DEAL\_AccountNumber\_\_c * Account.DEAL\_GUPNumber\_\_c   **Engagement:**   * Id * Name * DEAL\_Account\_\_r.Name * DEAL\_Account\_\_r.DEAL\_AccountNumber\_\_c * DEAL\_Account\_\_r.DEAL\_GUPNumber\_\_c   **Contact:**   * Id * Name * Account.Name * Account.DEAL\_AccountNumber\_\_c * Account.DEAL\_GUPNumber\_\_c |
| Lightning component | DEAL\_UploadingFiles |
| Visualforce | DEAL\_SharepointFrame |
| Apex class | DEAL\_UploadingFiles |

##### Email Notes

**Description:**  Regarding email templates, at the beginning we were using 2 separate email templates, one per German and one per English language. There are some mail automations that were left like this, but later in time we used the approach of the CZ team using substitute on email templates to replace dynamic fields, leave just one email template and handle the translation through the translation workbench. Maybe in a later point in time this can be changed to remove the approach of email template per language.

Also the CZ team neded more complex email templating, so they needed to have a visualforce component email template with only the body section of the mail, so most of email templates now consist of 3 components, the header, body and footer.

Regarding automation, we have email templates in workflows, process builder and in apex code. For the apex code we created a class to generate all of this, named Send Reminders Helper, that was later extended by the CZ team so it could be used to send mail related to any object.

**Medadata:**

|  |  |
| --- | --- |
| Type | Name |
| Visualforce component | DEAL\_EmailTemplatesHeader  DEAL\_EmailTemplatesBody  DEAL\_EmailTemplatesFooter |
| Email template | DEAL\_TemplateMaster |
| Apex class | DEAL\_SendRemindersHelper  DEAL\_SendRemindersBatch |