**CCO eCoaching DB Unit Test Document**

* **eCL surveys**

October 1, 2015

Change History Log

| Date | Revision | Change Description | Author |
| --- | --- | --- | --- |
| 09/28/2015 | 1.0 | TFS 549 – Initial revision. | Susmitha Palacherla |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1.0 TFS 549 – CSR Surveys 4](#_Toc429489072)

## TFS 549 – CSR Surveys

| Item | Description |
| --- | --- |
| Change Type | New Functionality |
| Change Description | Setup the database objects for generating Surveys based on completed ecls. This TFS work item is specific to the setup of the CSR Survey but unit testing will includedesign support for expanding to other surveys and modules as needed. |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated | 1. Tables  CREATE TABLE Survey\_DIM\_Type  CREATE TABLE Survey\_DIM\_Question  CREATE TABLE Survey\_DIM\_Response  CREATE TABLE Survey\_DIM\_QAnswer  CREATE TABLE Survey\_Response\_Header  CREATE TABLE Survey\_response\_Detail  2. Procedures  CREATE SP [EC].[sp\_InsertInto\_Survey\_Response\_Header]- SurveyGenerateInsertHeader and Expire  CREATE SP - NotificationSelect  CREATE SP- EmailSentUpdate  CREATE SP [EC].[sp\_Update\_Survey\_Response]- SurveyRespondUpdate  ALTER SPs - [EC].[sp\_Inactivations\_From\_Feed] AND [EC].[sp\_InactivateCoachingLogsForTerms] - Inactivate From feed and Terms  3. Notification Script  4. SSIS Package  5. SQL agent Job |
| Code doc | CCO\_eCoaching\_Surveys\_Create.sql |
| Notes | Useful sql  --delete from ec.Coaching\_Log\_reason  --where coachingid = 54876  --delete from ec.Coaching\_Log  --where formname = 'eCL-Kyle.Aldridge-54876'      --Delete from [EC].[Survey\_Response\_Header]  --where surveyid = 9    --DBCC CheckIdent("[eCoachingDev].[EC].[Survey\_Response\_Header]", RESEED,7)  -- Give the last id that remains in the table. |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 1.x | Test contents of survey Type Table  SELECT \* FROM [EC].[Survey\_DIM\_Type] | 1 row with CSR =1 | P | Other Modules should be 0.  For Unit testing design Supervisor or Other Module will be temporarily set to 1 and should be reset to 0 in run book. |
| 1.x | Test contents of Question DIM table  SELECT \* FROM [EC].[Survey\_DIM\_Question] | 5 questions and 1 row for Unknown Question value.  All Active. | P |  |
| 1.x | Test contents of Response DIM table  SELECT \* FROM [EC].[Survey\_DIM\_Response] | An ID for each possible Response possible for the survey. Responses in this table are not tied to a specific Question. Just a mapping of ID to value and Active indicator.  Should have a row for every possible response per requirements. | P |  |
| 1.x | Test contents of QAnswer DIM Table  SELECT \* FROM [EC].[Survey\_DIM\_QAnswer] | Unlike above Responses table this Table should have a row for every Question and Response combination possible with additional details like Response Type etc. | P |  |
| 1.x | Test Record select for Survey Generation per requirements  --Test data setup  --1. Select CSR records  SELECT \*  FROM [eCoachingDev].[EC].[Employee\_Hierarchy]  where Active = 'A'  and emp\_job\_code = 'WACS01'  order by Emp\_Name  GO  --2. Insert 2 or more records for some of the CSR lanids  USE [eCoachingDev]  GO  DECLARE @return\_value int,  @nvcNewFormName nvarchar(50)  EXEC @return\_value = [EC].[sp\_InsertInto\_Coaching\_Log]  @nvcFormName = N'Keta.Abner',  @nvcEmpLanID = N'Keta.Abner',  @nvcProgramName = N'Medicare',  @intSourceID = 102,  @intStatusID = 4,  @SiteID = NULL,  @nvcSubmitter = N'susmitha.palacherla',  @dtmEventDate = N'2015-09-06 14:08:00.000',  @dtmCoachingDate = N'2015-09-06 11:38:00.000',  @bitisAvokeID = 0,  @nvcAvokeID = NULL,  @bitisNGDActivityID = 0,  @nvcNGDActivityID = NULL,  @bitisUCID = 0,  @nvcUCID = NULL,  @bitisVerintID = 0,  @nvcVerintID = NULL,  @intCoachReasonID1 = 10,  @nvcSubCoachReasonID1 = N'42',  @nvcValue1 = N'Opportunity',  @intCoachReasonID2 = 13,  @nvcSubCoachReasonID2 = N'99',  @nvcValue2 = N'ReInforcement',  @intCoachReasonID3 =NULL,  @nvcSubCoachReasonID3 =NULL,  @nvcValue3 = NULL,  @intCoachReasonID4 = NULL,  @nvcSubCoachReasonID4 = NULL,  @nvcValue4 = NULL,  @intCoachReasonID5 = NULL,  @nvcSubCoachReasonID5 = NULL,  @nvcValue5 = NULL,  @intCoachReasonID6 = NULL,  @nvcSubCoachReasonID6 = NULL,  @nvcValue6 = NULL,  @intCoachReasonID7 = NULL,  @nvcSubCoachReasonID7 = NULL,  @nvcValue7 = NULL,  @intCoachReasonID8 = NULL,  @nvcSubCoachReasonID8 = NULL,  @nvcValue8 = NULL,  @intCoachReasonID9 = NULL,  @nvcSubCoachReasonID9 = NULL,  @nvcValue9 = NULL,  @intCoachReasonID10 = NULL,  @nvcSubCoachReasonID10 = NULL,  @nvcValue10 = NULL,  @intCoachReasonID11 = NULL,  @nvcSubCoachReasonID11 = NULL,  @nvcValue11 = NULL,  @intCoachReasonID12 = NULL,  @nvcSubCoachReasonID12 = NULL,  @nvcValue12 = NULL,  @nvcDescription = N'Test inserts for Survey generation.',  @nvcCoachingNotes = N'Test inserts for Survey generation.',  @bitisVerified = 1,  @dtmSubmittedDate = N'2015-09-06 11:39:00.000',  @dtmStartDate = N'2015-09-06 11:39:00.000',  @dtmSupReviewedAutoDate = NULL,  @bitisCSE = 0,  @dtmMgrReviewManualDate = NULL,  @dtmMgrReviewAutoDate = NULL,  @nvcMgrNotes = NULL,  @bitisCSRAcknowledged = NULL,  @dtmCSRReviewAutoDate = NULL,  @nvcCSRComments = NULL,  @bitEmailSent = 1,  @ModuleID = 1,  @Behaviour = NULL,  @nvcNewFormName = @nvcNewFormName OUTPUT  SELECT @nvcNewFormName as N'NewFormName'  SELECT 'Return Value' = @return\_value  GO  select \* from ec.Coaching\_Log  where coachingid =  order by emplanid      Update ec.Coaching\_Log  set isCSRAcknowledged = 1  ,CSRReviewAutoDate= ''  ,statusid = 1  where coachingid =  --and submitteddate = '2015-09-06 11:39:00.000'      update [EC].[Employee\_Hierarchy]  set [Emp\_Email]= 'susmitha.palacherla@gdit.com'  --,[Emp\_LanID]= ''  Where emp\_id = ''  --Execute to Insert or run sql agent job  and test results  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InsertInto\_Survey\_Response\_Header]  SELECT 'Return Value' = @return\_value  GO  SELECT [SurveyID]  ,[SurveyTypeID]  ,[CoachingID]  ,[FormName]  ,[EmpID]  ,[EmpLanID]  ,[SiteID]  ,[SourceID]  ,[ModuleID]  ,[CreatedDate]  ,[MonthOfYear]  ,[CalendarYear]  ,[CSRComments]  ,[EmailSent]  ,[CompletedDate]  ,[Status]  ,[InactivationDate]  ,[InactivationReason]  FROM [eCoachingDev].[EC].[Survey\_Response\_Header]  GO | 3 logs selected per test data that was setup.  Records were randomly selected. (the record with the earlier or later CSRReviewAutoDate was not consistent)  Repeated executions of SP did not insert additional records for same employee for the same month.  Only completed ecl logs were selected. | P |  |
| 1.x | Test Resend of a Survey  Can Test by either running  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InsertInto\_Survey\_Response\_Header\_Resend]  SELECT 'Return Value' = @return\_value  GO  Or by setting variable @Resend in ssis package to True and running package. | Should insert an additional survey for employees that already have one survey of that type for the month. | P |  |
| 1.x | Test Email Notification including EmailSent flag update  --Test Setup  Update all or specific employee logs that have surveys generated to have your email address. (Replace EmpIds in below sql to actual employees with pending surveys)  SELECT \* FROM [eCoachingDev].[EC].[Survey\_Response\_Header]  update [EC].[Employee\_Hierarchy]  set [Emp\_Email]= 'Lili.Huang@GDIT.com'  --,[Emp\_LanID]= 'Sherry.Abner'  Where emp\_id = '224459'  update [EC].[Employee\_Hierarchy]  set [Emp\_Email]= 'susmitha.palacherla@gdit.com'  --,[Emp\_LanID]= 'Sherry.Abner'  Where emp\_id in ( '243448','224459',  '244582',  '233527',  '221559','249230')  Update [eCoachingDev].[EC].[Survey\_Response\_Header]  set EmailSent = 0  where coachingid in ( 54863, 54698)  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectSurvey4Contact]  SELECT 'Return Value' = @return\_value  GO | Received notifications and EmailSent flag set to 1 | P |  |
| 1.x | Verify Email format and content per requirements. |  | P |  |
| 1.x | After Notifications are sent verify EmailSent is set to 1. | EmailSent should be set to 1 for all records previously selected for notification above. | P |  |
| 1.x | Test Record select for Survey Generation per requirements for additional module  Inserted ecls for Sup module and ensured they were in Completed state. | Survey was generated based on Supervisor being set to 1 in Survey Type DIM table. | P | \*Not part of this release but design support for future additions. |
| 1.x | Test Record select for Survey Generation per requirements for additional Survey Type | Pending Test |  | \*Not part of this release but design support for future additions. |
| 1.x | Pending surveys belonging to an employee are displayed for logged in employee  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Survey\_EmpPending]  @strEmpin = N'Kyle.Aldridge'  SELECT 'Return Value' = @return\_value  GO | Any pending Surveys for logged in user returned. | P |  |
| 1.x | Test Survey completion | Survey header record was updated and rows added to Response Detail table. | P | Several logs tested from Lilis local UI copy. |
| 1.x | Test Survey expiry after 5 days  SELECT \* FROM [eCoachingDev].[EC].[Survey\_Response\_Header]  where DATEDIFF(DAY, [CreatedDate], GETDATE())> 5  USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InactivateCoachingLogsForTerms]  SELECT 'Return Value' = @return\_value  GO | Survey status should get updated to ‘Inactive’  Inactivation date populated with test run datetime.  InactivationStatus ‘Survey Expired’ | P |  |
| 1.x | Test survey inactivation for terms  Find a survey log that is outside of expiry date range for inactive csr or temporarily setup csr to be Active = ‘T’  SELECT \* FROM [eCoachingDev].[EC].[Survey\_Response\_Header]  where DATEDIFF(DAY, [CreatedDate], GETDATE())< 5  and [Status] = 'Open'  Survey ID 17 found.  Set Emoployee to ‘T’  update ec.employee\_Hierarchy  set Active= 'T',  emp\_lanid = 'Felice.Abram'  , [End\_Date]= 20150930  where emp\_id = '220810'  USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InactivateCoachingLogsForTerms]  SELECT 'Return Value' = @return\_value  GO | Survey status should get updated to ‘Inactive’  Inactivation date populated with test run datetime.  InactivationStatus ‘Employee Not Active’ | P |  |
| 1.x | Test survey inactivation for in-activations from feed.  Setup file with ecl formname with an open survey  Run Coachinginactivations sql agent job. | Survey status should get updated to ‘Inactive’  Inactivation date populated with test run datetime.  InactivationStatus ‘eCL Inactivated’ | P |  |
| 1.x | Test ssis package flow |  | P |  |
| 1.x | Test sql agent job |  | P |  |
| 1.x |  |  |  |  |
| 1.x |  |  |  |  |
| 1.x |  |  |  |  |
| 1.x |  |  |  |  |