**CCO eCoaching DB Unit Test Document**

* **Admin Tool**

July 5, 2016

Change History Log

| Date | Revision | Change Description | Author |
| --- | --- | --- | --- |
| 6/1/2016 | 1.0 | TFS – 1709- Admin Tool setup | Susmitha Palacherla |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1. TFS2268 Quality Other Feed(s) - CTC Load 4](#_Toc448154700)

## TFS 1709 Admin Tool setup

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Setup admin tool to perform inactivations, recativations and reassignments |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated |  |
| Code doc | CCO\_eCoaching\_Admin\_Tool\_Create.sql- New CCO\_eCoaching\_Functions.sql - Updated  CCO\_eCoaching\_EmployeeHierarchy\_Load.sql- Updated  CCO\_eCoaching\_Log\_Create.sql- Updated |
| Notes | --example sql to use table datatype to test sp in backend  DECLARE @IDs [int\_udtt]  -- fill some values into that table  INSERT INTO @IDs VALUES(3), (5), (17), (42)  -- call your stored proc  DECLARE @return\_value int  EXEC    @return\_value = [prc\_Foo\_Delete]          @fooIds = @IDs,   -- pass in that UDT table type here!          @deleteReason = 2,          @comment = N'asfdasdf',          @deletedBy = N'asdfa'  SELECT  'Return Value' = @return\_value  GO  SELECT \* FROM [EC].[AT\_User]  WHERE [UserId]= '345712'  GO  UPDATE [EC].[AT\_User]  SET Active = 1  WHERE [UserId]= '345712'  INSERT INTO [EC].[AT\_User]  ([UserId],  [UserLanID],  [UserName],  [EmpJobCode],  [Active])  VALUES  ('345712','susmitha.palacherla', 'Palacherla, Susmitha','WISY13',1)    GO  SELECT \* FROM [EC].[AT\_Role]  SELECT \* FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]= '345712'  INSERT INTO [EC].[AT\_User\_Role\_Link]  ([UserId] ,  [RoleId])  VALUES  ('345712',101),  ('345712',103)  DELETE FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]= '345712'  AND RoleId =  SELECT \* FROM [EC].[AT\_Role\_Entitlement\_Link]  SELECT \* FROM [EC].[AT\_Action\_Reasons]  SELECT \* FROM [EC].[AT\_Module\_Access]  SELECT \* FROM [EC].[AT\_Role\_Access]  SELECT \* FROM [EC].[AT\_Reassign\_Status\_For\_Module]  SELECT \* FROM [eCoachingDev].[EC].[Coaching\_Log]  WHERE [EmpID] = '216933'  --WHERE CoachingID = 52977  SELECT [EmpID], COUNT(\*)  FROM [eCoachingDev].[EC].[Coaching\_Log]  WHERE [StatusID]<> 2  GROUP BY [EmpID]  HAVING COUNT(\*)>3  SELECT \* FROM [eCoachingDev].[EC].[Warning\_Log]  WHERE [EmpID] = '380017'  GO  SELECT [EmpID], COUNT(\*)  FROM [eCoachingDev].[EC].[Warning\_Log]  WHERE [StatusID]<> 2  GROUP BY [EmpID]  HAVING COUNT(\*)>3  update [eCoachingDev].[EC].[Warning\_Log]  set [StatusID]= 2  WHERE FormName = 'eCL-Doug.Stearns-5467'  update [eCoachingDev].[EC].[Coaching\_Log]  set [StatusID]= 2  WHERE FormName = 'eCL-sandra.torres-11444' |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 1.1 | User table is created and record inserted for current program mgr  SELECT \* FROM [EC].[AT\_User]  WHERE [UserId]= '500306'  GO | 1 row returned for user id | P |  |
| 1.2 | Role table with 4 roles inserted  SELECT \* FROM [EC].[AT\_Role] | Returns 4 rows | P |  |
| 1.3 | User role link table and entries for a user who is admin for Coaching and Warning  SELECT \* FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]= '500306' | 2 rows with the roleids for coaching admin and warning admin 101 and 103 | P |  |
| 1.4 | Role Entitlement link table  SELECT \* FROM [EC].[AT\_Role\_Entitlement\_Link]  where RoleId = 103 | Combination of entitlements for each roleid  101-201,202,203,204,206  102- 201,202,204  103 - 201,203,205,207 | P |  |
| 1.5 | Check entitlements for user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO  Repeat for diffrent user types: admins and non admins | Should return all rows as defined in role entitlement table and user role link table | P |  |
| 1.6 | Check Modules by Type for user type  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Modules\_By\_LanID]  @nvcEmpLanIDin = N'susmitha.palacherla',  @strTypein = N'Coaching'  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations for warnings and coaching and admin and non admin users for each |  | P |  |
| 1.7 | Check roles by user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Roles\_By\_User]  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO |  | P |  |
| 1.8 | Status by Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Status\_By\_Module]  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for remaining 4 modules (Ids 2 through 5) |  | P |  |
| 1.9 | Inactivate user list  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Employees\_Inactivation\_Reactivation]  @strRequesterLanId = N'Bryan.Crockett',  @strTypein = N'Coaching',  @strActionin = N'Inactivate',  @intModulein = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.10 | Inactivate reason list  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Action\_Reasons]  @strType = N'Coaching',  @strAction = N'Inactivate'  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.11 | Inactivate logs for user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strTypein = N'Coaching',  @strActionin = N'Inactivate',  @strEmployeein = N'223779',  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.12 | Audit record for inactivate  SELECT \*  FROM [EC].[AT\_Coaching\_Inactivate\_Reactivate\_Audit]  where FormName = 'eCL-nicholas.alfonso-674420'  GO |  | P |  |
| 1.13 |  |  |  |  |
| 1.14 | Reactivate user list  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Employees\_Inactivation\_Reactivation]  @strRequesterLanId = N'Bryan.Crockett',  @strTypein = N'Coaching',  @strActionin = N'Reactivate',  @intModulein = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.15 | Reactivate reason list  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Action\_Reasons]  @strType = N'Coaching',  @strAction = N'Reactivate'  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.16 | Reactivate logs for user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strTypein = N'Coaching',  @strActionin = N'Reactivate',  @strEmployeein = N'223779',  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.17 | Audit record for reactivate  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strTypein = N'Coaching',  @strActionin = N'Reactivate',  @strEmployeein = N'223779',  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO |  | P |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 1.18 | Reassign From  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignFrom\_Users]  @strRequesterin = N'susmitha.palacherla',  @intModuleIdin = 1,  @intStatusIdin = 5  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.19 | Reassign To  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignTo\_Users]  @strRequesterin = N'susmitha.palacherla',  @strFromUserIdin = N'alphonso.bell',  @intModuleIdin = 1,  @intStatusIdin = 5  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.20 | Reassign reason list  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Action\_Reasons]  @strType = N'Coaching',  @strAction = N'Reassign'  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.21 | Reassign logs for user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Reassign]  @istrOwnerin = N'234067',  @intStatusIdin = 5,  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 1.22 | Audit record for reassign  DECLARE @returnMessage nvarchar(100),  @returnCode int    DECLARE @tableIds1 AS EC.IdsTableType  INSERT @tableIds1  VALUES (27904),  (14850)  --, (49012)  --, (14849)  --EXEC [EC].[sp\_AT\_Coaching\_Inactivation\_Reactivation] N'johneric.tiongson',N'Reactivate',@tableIds1,N'Error in Submission',NULL, NULL, NULL  EXEC [EC].[sp\_AT\_Coaching\_Reassignment]  @strRequesterLanId = N'angela.reuter',  @tableIds= @tableIds1,  @strAssignedId = N'96653',  @intReasonId = 8,--Supervisor / manager unavailable  @strReasonOther = NULL,  @strComments = 'No Comments',  @returnCode = @returnCode OUTPUT,  @returnMessage = @returnMessage OUTPUT |  | P |  |
| 1.23 | Sup dashboard | Pending log - original owner – No  Pending log - reassigned owner – Yes  Completed log - original owner – Yes  Completed log - reassigned owner – Yes |  |  |
| 1.24 | Mgr Dashboard | Pending log – Yes with reassigned sup as owner |  |  |
| 1.25 | Review Page | 2 new attributes to display reassigned sup and reassigned mgr where applicable |  |  |
| 1.26 | Select logs for reassign when LCS | Display reviewing manager name |  |  |