**CCO eCoaching DB Unit Test Document**

* **Admin Tool**

January 19, 2018

Change History Log

| Date | Revision | Change Description | Author |
| --- | --- | --- | --- |
| 6/1/2016 | 1.0 | TFS – 1709- Admin Tool setup | Susmitha Palacherla |
| 9/9/2016 | 2.0 | TFS 3441 – Change Functionality in admin tool for inactive employees | Susmitha Palacherla |
| 9/20/2016 | 3.0 | TFS 3877 – Access for Mark Hackman and Scott Potter | Susmitha Palacherla |
| 10/26/2016 | 4.0 | TFS 4353 – Reset Reminder attributes on Reassign | Susmitha Palacherla |
| 03/03/2017 | 5.0 | TFS 5756 - Admin tool access for job code WPPM50 | Susmitha Palacherla |
| 04/11/2017 | 6.0 | TFS 6246 – Admin Tool access for Mark Hackman (job code WPSM13) | Susmitha Palacherla |
| 06/30/2017 | 7.0 | TFS 7152 – Allow for inactivation of Completed logs in Admin Tool | Susmitha Palacherla |
| 09/22/2017 | 8.0 | TFS 8363 - Set up admin tool access for new Program team members | Susmitha Palacherla |
| 01/04/2018 | 9.0 | TFS 7856 – Encryption of sensitive data | Susmitha Palacherla |

Contents

[1. TFS 1709 Admin Tool setup 4](#_Toc502902100)

[2. TFS 3441 Change Functionality in eCL Admin Tool for Inactive Employees 18](#_Toc502902101)

[3. TFS 3877 Admin Tool access for Mark Hackman and Scott Potter 30](#_Toc502902102)

[4. TFS 4353 Reset Reminder attributes on Reassign 32](#_Toc502902103)

[5. TFS 5756 Admin tool access for jobcode WPPM50 34](#_Toc502902104)

[6. TFS 6246 – Admin Tool access for Mark Hackman (job code WPSM13) 38](#_Toc502902105)

[7. TFS 7152 Allow inactivation of completed logs 40](#_Toc502902106)

[8. TFS 8363 – Set up admin tool access for new Program team members (job code WPPM11) 44](#_Toc502902107)

[9. TFS 7856 Encryption of sensitive data 46](#_Toc502902108)

## 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].[Employee\_Hierarchy]  WHERE [Emp\_ID]= '345712'  update [EC].[Employee\_Hierarchy]  set Emp\_Job\_Code = 'wacs50'  WHERE [Emp\_ID]= '345712'  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]  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO  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 |  |  |

## TFS 3441 Change Functionality in eCL Admin Tool for Inactive Employees

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Change functionality for eCL Admin Tool for inactive employee, supervisor, manager |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated |  |
| Code doc | CCO\_eCoaching\_Admin\_Tool\_Create.sql-  CCO\_eCoaching\_EmployeeHierarchy\_Load.sql |
| Notes | SELECT DISTINCT Emp.Emp\_ID,Emp.Emp\_Name, Emp\_Site  FROM [EC].[Employee\_Hierarchy] Emp JOIN [EC].[Warning\_Log] Fact WITH(NOLOCK)  ON Emp.Emp\_ID = Fact.EmpID  WHERE Fact.StatusID = 1  AND Emp.Active IN ('P','L')  set Emp\_Job\_Code = 'wacs50'  WHERE [Emp\_ID]= '345712'  Emp\_ID Emp\_Name Emp\_Site  228333 Hansen, Merlinda Sandy  246246 Reynoso, Carolina Sandy  246387 Munoz, Isaac R Sandy  SELECT DISTINCT Emp.Emp\_ID,Emp.Emp\_Name, Emp\_Site  FROM [EC].[Employee\_Hierarchy] Emp JOIN [EC].[Coaching\_Log] Fact WITH(NOLOCK)  ON Emp.Emp\_ID = Fact.EmpID  WHERE Fact.StatusID not in (1,2)  AND Emp.Active IN ('P','L')  and Emp\_Site = 'Hattiesburg'  and Fact.ModuleID = 2  228659 Frank, Cotina A Hattiesburg  Below for CSR Module at Hattiesburg inactivation  269274 Woods, TaKara D Hattiesburg  244563 Hughes, Angela D Hattiesburg  254678 Barnes, Angelica S Hattiesburg  271658 Pollock, Clifton J Hattiesburg  279228 Sullivan, Qiana T Hattiesburg |
|  |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 2.1 | Inactivate Coaching Logs  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  Pick an employee from above result set.  Update Active status in Hierarchy table to each of the 5 statuses.  -- 280253  Aguayo Romero, Carlos D  update EC.Employee\_Hierarchy  set Active = 'T'  where Emp\_ID = '280253' | A – Appear in drop down  T – Do not appear in drop down  D – Do not appear in drop down  L – Appear in drop down  P – Appear in drop down | P  P  P  P  P |  |
| 2.2 | Inactivate Warning Logs  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Employees\_Inactivation\_Reactivation]  @strRequesterLanId = N'Bryan.Crockett',  @strTypein = N'Warning',  @strActionin = N'Inactivate',  @intModulein = 1  SELECT 'Return Value' = @return\_value  GO  Pick an employee from above result set.  Update Active status in Hierarchy table to each of the 5 statuses.  -- 424339  Darton, Jacklyn C  update EC.Employee\_Hierarchy  set Active = 'T'  where Emp\_ID = '424339' | A – Appear in drop down  T – Do not appear in drop down  D – Do not appear in drop down  L – Appear in drop down  P – Appear in drop down | P  P  P  P  P |  |
| 2.3 | Inactivate Surveys | N/A | N/A | Functionality not available in admin tool |
| 2.4 | Reassign Logs From Sup/Mgr  Pending Manager review  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignFrom\_Users]  @strRequesterin = N'Bryan.Crockett',  @intModuleIdin = 1,  @intStatusIdin = 5  SELECT 'Return Value' = @return\_value  GO  Pick an employee from above result set.  Update Active status in Hierarchy table to each of the 5 statuses.  -- 222462  Dron, Annette  update EC.Employee\_Hierarchy  set Active = 'L'  where Emp\_ID = '222462'  Repeat for Pending sup review  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignFrom\_Users]  @strRequesterin = N'Bryan.Crockett',  @intModuleIdin = 1,  @intStatusIdin = 6  SELECT 'Return Value' = @return\_value  GO  Pick an employee from above result set.  Update Active status in Hierarchy table to each of the 5 statuses.  -- 222535  Collins, Tamara  update EC.Employee\_Hierarchy  set Active = 'L'  where Emp\_ID = ' 222535' | A – Appear in drop down  T – Appear in drop down  D Appear in drop down  L – Appear in drop down  P – Appear in drop down  A – Appear in drop down  T – Appear in drop down  D Appear in drop down  L – Appear in drop down  P – Appear in drop down | P  P  P  P  P  P  P  P  P  P |  |
| 2.5 | Reassign Logs To Sup/Mgr  Pending Manager Review  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  262789 Campbell, John W  update EC.Employee\_Hierarchy  set Active = 'A'  where Emp\_ID = '262789'  Repeat for all values of Active.  Then Repeat test for pending sup review  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignTo\_Users]  @strRequesterin = N'susmitha.palacherla',  @strFromUserIdin = N'alphonso.bell',  @intModuleIdin = 1,  @intStatusIdin = 6  343925 Cuba, Sonia Y  update EC.Employee\_Hierarchy  set Active = 'A'  where Emp\_ID = '343925'  SELECT 'Return Value' = @return\_value  GO | A – Appear in drop down  T – Do not appear in drop down  D Do not appear in drop down  L – Do not appear in drop down  P – Do not appear in drop down  A – Appear in drop down  T – Do not appear in drop down  D Do not appear in drop down  L – Do not appear in drop down  P – Do not appear in drop down | P  P  P  P  P  P  P  P  P  P |  |
| 2.6 | Reassign Coaching Emp Logs in pending Mgr Review  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  Martin, Brandon I  update EC.Employee\_Hierarchy  set Active = 'L'  where Emp\_Name = 'Martin, Brandon I'  Reassign Coaching Emp Logs in pending Sup Review  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Reassign]  @istrOwnerin = N'234067',  @intStatusIdin = 6,  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Aiken, Crystal L  update EC.Employee\_Hierarchy  set Active = 'L'  where Emp\_Name = 'Aiken, Crystal L' | A – Appear in drop down  T – Do not appear in drop down  D – Do not appear in drop down  L – Appear in drop down  P – Appear in drop down  A – Appear in drop down  T – Do not appear in drop down  D – Do not appear in drop down  L – Appear in drop down  P – Appear in drop down | P  P  P  P  P  P  P  P  P  P |  |
| 2.7 | Reactivate Coaching Logs  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  Pick an employee from above result set.  Update Active status in Hierarchy table to each of the 5 statuses.  -- 256496  256496 Aberdeen, Kerwin D  update EC.Employee\_Hierarchy  set Active = 'A'  where Emp\_ID = '256496' | A – Appear in drop down  T – Do not appear in drop down  D Do not appear in drop down  L – Do not appear in drop down  P – Do not appear in drop down | P  P  P  P  P |  |
| 2.8 | Reactivate Warning Logs  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Employees\_Inactivation\_Reactivation]  @strRequesterLanId = N'Bryan.Crockett',  @strTypein = N'Warning',  @strActionin = N'Reactivate',  @intModulein = 1  SELECT 'Return Value' = @return\_value  GO  Pick an employee from above result set.  Update Active status in Hierarchy table to each of the 5 statuses.  -- 231884 Rogers, Lydia S  update EC.Employee\_Hierarchy  set Active = 'A'  where Emp\_ID = '231884' | A – Appear in drop down  T – Do not appear in drop down  D Do not appear in drop down  L – Do not appear in drop down  P – Do not appear in drop down | P  P  P  P  P |  |
| 2.9 | Reactivate Surveys | N/A | N/A | Functionality not available in admin tool |
| 2.10 | Employee Feed function  Inactive Coaching Logs  SELECT cl.\*  FROM [EC].[Coaching\_Log]cl join EC.Employee\_Hierarchy eh  on cl.EmpID = eh.Emp\_ID  where cl.StatusID not in (1,2)  and eh.Active = 'A'  and SiteID = 5  and CoachingID > 50000  GO   1. 341163 keep 'A' 2. 234484 update to 'T' eCL-jennifer.goodman-50211 - Should change to 2   update [EC].[Employee\_Hierarchy]  set Active = 'T'  ,[End\_Date]= '20160909'  Where Emp\_ID = '234484'   1. 219447 update to 'D'   update [EC].[Employee\_Hierarchy]  set Active = 'D'  ,[End\_Date]= '20160909'  Where Emp\_ID = '219447'   1. 255123 update to 'L' update [EC].[Employee\_Hierarchy]   set Active = 'L'  ,[End\_Date]= '20160909'  Where Emp\_ID = '255123'   1. 235719 update to 'P'   update [EC].[Employee\_Hierarchy]  set Active = 'P'  ,[End\_Date]= '20160909'  Where Emp\_ID = '235719'  select \* from [EC].[Coaching\_Log]cl  where FormName in (  'eCL-frederick.stieglitz-50133',  'eCL-jennifer.goodman-50211',  'eCL-jacqueline.henry-50254',  'eCL-michael.l.cherry-50426',  'eCL-sue.steele-50496'  ) | Ecl-frederick.stieglitz-50133 should remain at 3  eCL-jennifer.goodman-50211 - Should change to 2  eCL-jacqueline.henry-50254 - Should change to 2  eCL-michael.l.cherry-50426 - should remain at 6  eCL-sue.steele-50496 - should remain at 3 | P  P  P  P  P |  |
| 2.11 | Inactive Warning Logs  SELECT cl.\*  FROM [EC].[Warning\_Log]cl join EC.Employee\_Hierarchy eh  on cl.EmpID = eh.Emp\_ID  where cl.StatusID <> 2  and eh.Active = 'A'   1. 409509 keep 'A' - 2. 218825 update to 'T' -   update [EC].[Employee\_Hierarchy]  set Active = 'T'  ,[End\_Date]= '20160909'  Where Emp\_ID = '218825'   1. 223651 update to 'D'   update [EC].[Employee\_Hierarchy]  set Active = 'D'  ,[End\_Date]= '20160909'  Where Emp\_ID = '223651'  4.225390 update to 'L'  update [EC].[Employee\_Hierarchy]  set Active = 'L'  ,[End\_Date]= '20160909'  Where Emp\_ID = '225390'  5.225735 update to 'P'  update [EC].[Employee\_Hierarchy]  set Active = 'P'  ,[End\_Date]= '20160909'  Where Emp\_ID = '225735'  select \* from [EC].[Warning\_Log]  where FormName in (  'eCL-Endalkachew.Negussie-11',  'eCL-Corey.Hinton-403133',  'eCL-justin.kramer-5358',  'eCL-dawn.kasper-5317',  'eCL-liana.vasquez-5371'  ) | eCL-Endalkachew.Negussie-11 should remain at 1  eCL-Corey.Hinton-403133 - Should remain at 1  eCL-justin.kramer-5358 - Should change to 2  eCL-dawn.kasper-5317- should remain at 1  eCL-liana.vasquez-5371 - should remain at 1 | P  P  P  P  P |  |

## TFS 3877 Admin Tool access for Mark Hackman and Scott Potter

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Setup admin tool access to Mark Hackman and Scott Potter |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated | Table [EC].[AT\_User]  Table [EC].[AT\_User\_Role\_Link]  Table [EC].[AT\_Role\_Access] |
| Code doc | CCO\_eCoaching\_Admin\_Tool\_Create.sql  CCO\_eCoaching\_AT\_Dimension\_Table\_Data.xlsx |
| Notes |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 3.1 | SELECT \* FROM [EC].[AT\_User]  WHERE [UserId]in ('343549','408246')  GO | Rows returned for Mark and scott | P |  |
| 3.2 | SELECT \* FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]in ('343549','408246') | 2 rows with the rolei ds for coaching admin and warning admin 101 and 103 returned for each user | P |  |
| 3.3 | SELECT \* FROM [EC].[AT\_Role\_Access]  WHERE JobCode = 'WACQ13' | 2 rows returned for the job code WACS13 for each of the roles 101 and 103. | P |  |
| 3.4 | Check entitlements for users  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'Scott.Potter'  SELECT 'Return Value' = @return\_value  GO    Repeat test for second user  Mark.Hackman | Should return all rows as defined in role entitlement table and user role link table  EntitlementId EntitlementDescription  201 EmployeeLog-SearchForInactivate  202 EmployeeLog-SearchForReassign  203 EmployeeLog-SearchForReactivate  204 ManageCoachingLogs  205 ManageWarningLogs  206 ReactivateCoachingLogs  207 ReactivateWarningLogs | P |  |
| 3.5 | Check Modules for users  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Modules\_By\_LanID]  @nvcEmpLanIDin = N'Scott.Potter',  @strTypein = N'Coaching'  SELECT 'Return Value' = @return\_value  GO Repeat test for second user  Mark.Hackman | ModuleId Module  1 CSR  4 LSA  3 Quality  2 Supervisor  5 Training | P |  |
| 3.6 | Check roles by user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Roles\_By\_User]  @nvcEmpLanIDin = N'Scott.Potter'  SELECT 'Return Value' = @return\_value  GO | UserId RoleDescription  408246 CoachingAdmin  408246 WarningAdmin | P |  |
|  |  |  |  |  |
|  |  |  |  |  |

## TFS 4353 Reset Reminder attributes on Reassign

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Updating recipients for reminders.  Realized that the reminder count is 2 per reviewer/status, so reminder attributes will need to be reset on reassign of logs |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated | [EC].[sp\_AT\_Coaching\_Reassignment] |
| Code doc | CCO\_eCoaching\_Admin\_Tool\_Create.sql |
| Notes |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 4.1 | Verify that Reminder Count , Reminder date and Reminder Sent flag get reset on reassign  Find a log that has a reminder count and reassign  DECLARE @returnMessage nvarchar(100),  @returnCode int    DECLARE @tableIds1 AS EC.IdsTableType  INSERT @tableIds1  VALUES (41595)  --, (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'johneric.tiongson',  @tableIds= @tableIds1,  @strAssignedId = N'91885',  @intReasonId = 8,--Supervisor / manager unavailable  @strReasonOther = NULL,  @strComments = 'No Comments',  @returnCode = @returnCode OUTPUT,  @returnMessage = @returnMessage OUTPUT | Reminder Count , Reminder date and Reminder Sent flag get reset on reassign | *P* |  |
|  |  |  |  |  |

## TFS 5756 Admin tool access for jobcode WPPM50

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Program has requested to grant admin tool access to staff with job code WPPM50.  (access level similar to WACS50/60) |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated | Added rows to tables  Table [EC].[AT\_Module\_Access]  Table [EC].[AT\_Role\_Access] |
| Code/Data doc | CCO\_eCoaching\_AT\_Dimension\_Table\_Data.xlsx |
| Notes | SELECT \*  FROM [eCoachingDev].[EC].[Employee\_Hierarchy]  where (Emp\_Job\_Code = 'wppm50' or emp\_id = '345712')  and Active = 'A'  GO  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*    --update [eCoachingDev].[EC].[Employee\_Hierarchy]  --set Emp\_Job\_Code = 'wppm50'  --,Emp\_Job\_Description = 'Manager, Program'  --where emp\_id = '345712'  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*    SELECT \*  FROM [eCoachingDev].[EC].[AT\_User]  WHERE Active = 1  and [EmpJobCode] = 'wppm50'  --and userid = '345712'  GO  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*  SELECT \*  FROM [eCoachingDev].[EC].[AT\_Role\_Access]  where JobCode in ('WACS50' ,'WACS60','WPPM50')  GO  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*  SELECT \*  FROM [eCoachingDev].[EC].[AT\_Module\_Access]  WHERE JobCode in ('WACS50' ,'WACS60','WPPM50')  GO  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*    --DECLARE @return\_value int  --EXEC @return\_value = [EC].[sp\_AT\_Populate\_User]  --SELECT 'Return Value' = @return\_value  --GO  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'Susmitha.Palacherla'  SELECT 'Return Value' = @return\_value  GO  --\*\*\*\*\*\*\*\*\*\*\*\*\*\*    SELECT \*  FROM [eCoachingDev].[EC].[AT\_User\_Role\_Link]  GO    --\*\*\*\*\*\*\*\*\*\*\*\*\*\* |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 5.1 | Check users are added to User table after running AT\_populate\_User sp  SELECT \* FROM [EC].[AT\_User]  WHERE [UserId]in (SELECT Emp\_ID  FROM [eCoachingDev].[EC].[Employee\_Hierarchy]  where Emp\_Job\_Code = 'wppm50'  and Active = 'A'  )  GO | 11 users should be returned  (10 actual WPPM50 users and one my record updated for testing) | P |  |
|  |  |  |  |  |
| 5.2 | Check that User Role Links are created  SELECT \* FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]in (SELECT Emp\_ID  FROM [eCoachingDev].[EC].[Employee\_Hierarchy]  where Emp\_Job\_Code = 'wppm50'  and Active = 'A') | 102 role for users | P |  |
|  |  |  |  |  |
| 5.3 | Check Role access table  SELECT \* FROM [EC].[AT\_Role\_Access]  WHERE JobCode = 'WPPM50' | CoachingUser | P |  |
|  |  |  |  |  |
| 5.4 | Check Module access for users  SELECT \* FROM [EC].[AT\_Module\_Access]  WHERE JobCode = 'WPPM50' | CSR  Supervisor | P |  |
|  |  |  |  |  |
| 5.5 | Check entitlement for users  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'juan.chalarca'  SELECT 'Return Value' = @return\_value  GO | EntitlementId EntitlementDescription  201 EmployeeLog-SearchForInactivate  202 EmployeeLog-SearchForReassign  204 ManageCoachingLogs | P |  |
|  |  |  |  |  |
| 5.5 | Check access from UI  <https://f3420-mpmd01.vangent.local/eclAdmin_dev/> | Admin tool home page with option to manage logs | P |  |
|  |  |  |  |  |
| 5.6 | Check functionality from UI  Inactivate logs | Log should be inactivated successfully | P | eCL-cortnie.wright-52139 |
|  |  |  |  |  |
| 5.7 | Check functionality from UI  Reassign logs | Log should be reassigned successfully | P | eCL-pushpanand.mandal-54756 |
|  |  |  |  |  |

## TFS 6246 – Admin Tool access for Mark Hackman (job code WPSM13)

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Setup admin tool access to Mark Hackman = job code WPSM13 |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated | Table [EC].[AT\_User]  Table [EC].[AT\_User\_Role\_Link]  Table [EC].[AT\_Role\_Access] |
| Code doc | CCO\_eCoaching\_Admin\_Tool\_Create.sql  CCO\_eCoaching\_AT\_Dimension\_Table\_Data.xlsx |
| Notes |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 6.1 | SELECT \* FROM [EC].[AT\_User]  WHERE [UserId]='343549'  GO | Rows returned for Mark SHOULD HAVE Active = 1 | P |  |
| 6.2 | SELECT \* FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]in ('343549','408246') | 2 rows with the rolei ds for coaching admin and warning admin 101 and 103 returned for each user | P |  |
| 6.3 | SELECT \* FROM [EC].[AT\_Role\_Access]  WHERE JobCode = 'WACQ13' | 2 rows returned for the job code WPSM13 for each of the roles 101 and 103. | P |  |
| 6.4 | Check entitlements for users  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'Scott.Potter'  SELECT 'Return Value' = @return\_value  GO    Repeat test for second user  Mark.Hackman | Should return all rows as defined in role entitlement table and user role link table  EntitlementId EntitlementDescription  201 EmployeeLog-SearchForInactivate  202 EmployeeLog-SearchForReassign  203 EmployeeLog-SearchForReactivate  204 ManageCoachingLogs  205 ManageWarningLogs  206 ReactivateCoachingLogs  207 ReactivateWarningLogs | P |  |
| 6.5 | Check Modules for user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Modules\_By\_LanID]  @nvcEmpLanIDin = N'Mark.Hackman',  @strTypein = N'Coaching'  SELECT Return\_Value = @return\_value | ModuleId Module  1 CSR  4 LSA  3 Quality  2 Supervisor  5 Training | P |  |
| 6.6 | Check roles by user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Roles\_By\_User]  @nvcEmpLanIDin = N'Scott.Potter'  SELECT 'Return Value' = @return\_value  GO | UserId RoleDescription  408246 CoachingAdmin  408246 WarningAdmin | P |  |
|  |  |  |  |  |
|  |  |  |  |  |

## TFS 7152 Allow inactivation of completed logs

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Change functionality for completed logs to be inactivated |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated |  |
| Code doc | sp\_AT\_Select\_Employees\_Coaching\_Inactivation\_Reactivation.sql  sp\_AT\_Select\_Logs\_Inactivation\_Reactivation.sql |
| Notes | -- Only completed  select distinct emp\_id, emp\_name  from ec.Employee\_Hierarchy eh join EC.Coaching\_Log ec  on eh.Emp\_ID = ec.EmpID  and ec.StatusID = 1  and eh.Active not in ('T','D')  --and eh.Emp\_Job\_Code = 'wacs40'  --order by eh.Emp\_Name  -- Everything besides completed and inactive  Except  select distinct emp\_id, emp\_name  from ec.Employee\_Hierarchy eh join EC.Coaching\_Log ec  on eh.Emp\_ID = ec.EmpID  and ec.StatusID not in (1,2)  and eh.Active not in ('T','D')  --and eh.Emp\_Job\_Code = 'wacs40'  order by eh.Emp\_Name  -\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  --WHERE (Fact.StatusID = 1 AND Fact.SubmittedDate > DATEADD(MM,-4, GETDATE()))  -\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  select \* from EC.Coaching\_Log ec  where ec.EmpID in ('246641',  '365226')  and ec.SubmittedDate > DATEADD(MM,-3, GETDATE())  -\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  EXEC [EC].[sp\_AT\_Select\_Employees\_Inactivation\_Reactivation]  @strRequesterLanId = N'johneric.tiongson',  @strTypein = N'Coaching',  @strActionin = N'Inactivate',  @intModulein = 1 |
|  |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 7.1 | Find an active Employee with Completed logs | Appear in drop down in admin tool for coaching admin only and only users who have completed logs in last 3 months.  Crystal.Alexander should appear in dropdown  Lili.huamg should not appear although she has completed logs only (they are older than 3 months) | P | 246641  Crystal.Alexander |
| 7.2 | Look at the logs returned for the employee | Completed logs should be displayed only if submitted within last 3 months  eCL-Crystal.Alexander-66857 | P |  |
| 7.3 | Logs returned for Charles as coaching admin  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strRequesterLanId = N'susmitha.palacherla',  @strTypein = N'Coaching',  @strActionin = N'Inactivate',  @strEmployeein = N'226363',  @intModuleIdin = 2  SELECT 'Return Value' = @return\_value  GO | Completed returned  eCL-charles.derusso-66955  eCL-charles.derusso-66952  eCL-charles.derusso-66949 |  | 226363  charles.derusso  Phoenix |
| 7.4 | Logs returned for Charles as non coaching admin  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strRequesterLanId = N'Victor.Hernandez',  @strTypein = N'Coaching',  @strActionin = N'Inactivate',  @strEmployeein = N'226363',  @intModuleIdin = 2  SELECT 'Return Value' = @return\_value  GO | Completed not returned  eCL-charles.derusso-66955  eCL-charles.derusso-66952 |  | Phx (siteid 4 mgrs)  Victor.Hernandez  Pilar.Gomez  theresa.rufrano  theresa.rufrano  tony.hernandez |
| 7.5 | Inactivate a completed log | Should inactivate successfully | P |  |
| 7.6 | Reactivate an inactive log with lastknownstatus 2 | Should be activated and back in Completed status | P |  |
|  |  |  |  |  |

## TFS 8363 – Set up admin tool access for new Program team members (job code WPPM11)

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Setup admin tool access to Mark Hackman = job code WPSM13 |
| Test Environment | eCoaching\_Dev database on F3420-ECLDBD01 |
| Code Modules created/updated | Table [EC].[AT\_User]  Table [EC].[AT\_User\_Role\_Link]  Table [EC].[AT\_Role\_Access] |
| Code doc | CCO\_eCoaching\_Admin\_Tool\_Create.sql  CCO\_eCoaching\_AT\_Dimension\_Table\_Data.xlsx |
| Notes |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 8.1 | SELECT \* FROM [EC].[AT\_User]  WHERE [UserId] in  (  '333386',  '397938'  )  GO | Rows returned for both users SHOULD HAVE Active = 1 | P |  |
| 8.2 | SELECT \* FROM [EC].[AT\_User\_Role\_Link]  WHERE [UserId]in  (  '333386',  '397938'  )  GO | 2 rows with the roleids for coaching admin and warning admin 101, 103,106 and 107 returned for each user | P | 333386 101  333386 103  333386 106  333386 107  397938 101  397938 103  397938 106  397938 107 |
| 8.3 | SELECT \* FROM [EC].[AT\_Role\_Access]  WHERE JobCode = 'WPPM11' | 4 rows returned for the job code WPPM11 for each of the roles 101, 103,106 and 107 | P |  |
| 8.4 | Check entitlements for users  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'Sara.Stonecipher'  SELECT 'Return Value' = @return\_value  GO    Repeat test for second user  shelly.encke | Should return all rows as defined in role entitlement table and user role link table  EntitlementId EntitlementDescription  201 EmployeeLog-SearchForInactivate  202 EmployeeLog-SearchForReassign  203 EmployeeLog-SearchForReactivate  204 ManageCoachingLogs  205 ManageWarningLogs  206 ReactivateCoachingLogs  207 ReactivateWarningLogs  209 Reports  210 Report-RunCoachingSummary  211 Report-RunWarningSummary  212 Report-RunHierarchySummary  213 Report-RunAdminActivitySummary  215 Users  216 User-eCoachingAccessControlList | P |  |
| 8.5 | Check Modules for user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Modules\_By\_LanID]  @nvcEmpLanIDin = N'shelly.encke',  @strTypein = N'Coaching'  SELECT Return\_Value = @return\_value  Repeat for  Sara.Stonecipher | ModuleId Module  1 CSR  4 LSA  3 Quality  2 Supervisor  5 Training | P |  |
| 8.6 | Check roles by user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Roles\_By\_User]  @nvcEmpLanIDin = N'Sara.Stonecipher'  SELECT 'Return Value' = @return\_value  GO  Repeat test for second user  shelly.encke | UserId RoleDescription  397938 CoachingAdmin  397938 WarningAdmin  397938 ReportCoachingAdmin  397938 ReportWarningAdmin | P |  |
|  |  |  |  |  |
|  |  |  |  |  |

## TFS 7856 Encryption of sensitive data

| Item | Description |
| --- | --- |
| Change Type | Change Request |
| Change Description | Encryption of sensitive data |
| Test Environment | eCoachingDev database on f3420-ecldbd01 |
| Code Modules created/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].[Employee\_Hierarchy]  WHERE [Emp\_ID]= '345712'  update [EC].[Employee\_Hierarchy]  set Emp\_Job\_Code = 'wacs50'  WHERE [Emp\_ID]= '345712'  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]  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Check\_Entitlements]  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO  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 =  update [eCoachingDev].[EC].[Coaching\_Log]  set [StatusID]= 2  WHERE FormName = |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 9.1 | User names are encrypted in table  SELECT \* FROM [EC].[AT\_User]  Sql for comparing old values  Select UserLanID\_Drop AS [Original LanID],  CONVERT(nvarchar(30),DecryptByKey(UserLanID)) AS [Decrypted LanID],  UserName\_Drop AS [Original Name],  CONVERT(nvarchar(50),DecryptByKey(UserName)) AS [Decrypted Name]  FROM EC.AT\_User | Column(s) Encrypted  UserName  UserLanID | P |  |
| 9.2 | 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 |  |
| 9.3 | 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 | ModuleId Module  1 CSR  4 LSA  3 Quality  2 Supervisor  5 Training | P |  |
| 9.4 | 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 | UserId RoleDescription  345712 CoachingAdmin  345712 WarningAdmin  345712 ReportWarningAdmin  345712 ReportCoachingCSRUser  345712 ACLAdmin | P |  |
| 9.5 | Inactivate user list  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Employees\_Coaching\_Inactivation\_Reactivation]  @strRequesterLanId = N'johneric.tiongson',  @strActionin = N'Inactivate',  @intModulein = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters | All employees with active coaching logs at all sites | P |  |
| 9.6 | Inactivate logs for user  OPEN SYMMETRIC KEY [CoachingKey]  DECRYPTION BY CERTIFICATE [CoachingCert]  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strRequesterLanId = N'johneric.tiongson',  @strTypein = N'Coaching',  @strActionin = N'Inactivate',  @strEmployeein = N'216933',  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters | Active logs for user | P |  |
| 9.7 | Inactivate log  DECLARE @return\_value int,  @returnCode int,  @returnMessage varchar(100)  DECLARE @tableIds1 AS EC.IdsTableType  INSERT @tableIds1  VALUES (65841)    EXEC @return\_value = [EC].[sp\_AT\_Coaching\_Inactivation\_Reactivation]  @strRequesterLanId = N'Sara.Stonecipher',  @strAction = N'Inactivate',  @tableIds = @tableIds1,  @intReasonId = 1,  @strReasonOther = NULL,  @strComments = NULL,  @returnCode = @returnCode OUTPUT,  @returnMessage = @returnMessage OUTPUT | sp\_AT\_Coaching\_Inactivation\_Reactivation completed successfully |  |  |
| 9.8 | Audit record for inactivate  SELECT \*  FROM [EC].[AT\_Coaching\_Inactivate\_Reactivate\_Audit]  where coachingid = 65841  GO | Log returned | P |  |
| 9.9 | Select Employees for warning DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Employees\_Warning\_Inactivation\_Reactivation]  @strRequesterLanId = N'johneric.tiongson',  @strActionin = N'Inactivate',  @intModulein = 1  SELECT 'Return Value' = @return\_value  GO | Users with active warning logs returned |  |  |
| 9.10 | Select warning logs for user  OPEN SYMMETRIC KEY [CoachingKey]  DECRYPTION BY CERTIFICATE [CoachingCert]  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strRequesterLanId = N'johneric.tiongson',  @strTypein = N'Warning',  @strActionin = N'Inactivate',  @strEmployeein = N'279304',  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO | Logs returned. |  |  |
| 9.11 | 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 | Users with logs for reactivation | P |  |
| 9.12 | 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 |  |
| 9.13 | Reactivate logs for user  OPEN SYMMETRIC KEY [CoachingKey]  DECRYPTION BY CERTIFICATE [CoachingCert]  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Logs\_Inactivation\_Reactivation]  @strRequesterLanId = N'johneric.tiongson',  @strTypein = N'Coaching',  @strActionin = N'Reactivate',  @strEmployeein = N'216933',  @intModuleIdin = 1  SELECT 'Return Value' = @return\_value  GO  Repeat for combinations of parameters |  | P |  |
| 9.14 | Reassign From  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignFrom\_Users]  @strRequesterin = N'johneric.tiongson',  @intModuleIdin = 1,  @intStatusIdin = 6  SELECT 'Return Value' = @return\_value  GO | List of users to reassign logs from | P |  |
| 9.15 | Reassign To  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_ReassignTo\_Users]  @strRequesterin = N'johneric.tiongson',  @strFromUserIdin = N'335034',  @intModuleIdin = 1,  @intStatusIdin = 6  SELECT 'Return Value' = @return\_value  GO | List of users to reassign logs to | P |  |
| 9.16 | 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 |  |
| 9.17 | 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 |  |
| 9.18 | Audit record for reassign  DECLARE @returnMessage nvarchar(100),  @returnCode int    DECLARE @tableIds1 AS EC.IdsTableType  INSERT @tableIds1  VALUES (55373)  --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 |  |
| 9.19 | Roles by user  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_AT\_Select\_Roles\_By\_User]  @nvcEmpLanIDin = N'johneric.tiongson'  SELECT 'Return Value' = @return\_value  GO | UserId RoleDescription  500306 CoachingAdmin  500306 WarningAdmin  500306 ReportCoachingAdmin  500306 ReportWarningAdmin  500306 ACLAdmin | P |  |
| 9.20 | Check agent role  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Check\_AgentRole]  @nvcLANID = N'johneric.tiongson',  @nvcRole = N'ecl'  SELECT 'Return Value' = @return\_value  GO | 168 | P |  |
| 9.21 | Check app role  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Check\_AppRole]  @nvcLANID = N'johneric.tiongson'  SELECT 'Return Value' = @return\_value  GO | Y | P |  |
| 9.22 | Sup dashboard | Pending log - original owner – No  Pending log - reassigned owner – Yes  Completed log - original owner – Yes  Completed log - reassigned owner – Yes | P |  |
| 9.23 | Mgr Dashboard | Pending log – Yes with reassigned sup as owner | P |  |
| 9.24 | Review Page | 2 new attributes to display reassigned sup and reassigned mgr where applicable | P |  |
| 9.25 | Select logs for reassign when LCS | Display reviewing manager name | P |  |