**CCO eCoaching DB Unit Test Document**

* **Warnings in eCL**

March 24, 2020

Change History Log

| Date | Revision | Change Description | Author |
| --- | --- | --- | --- |
| 10/03/2014 | 1.0 | SCR – 3479 – Incorporate CSR warnings in eCL | Susmitha Palacherla |
| 10/22/2014 | 1.1 | SCR - 13479 – Requirement changes – Additional testing | Susmitha Palacherla |
| 10/28/2014 | 2.0 | SCR – 13624 – Inactivate expired warnings | Susmitha Palacherla |
| 07/23/2015 | 3.0 | TFS 363/402 – Fix issue with Duplicate Form names for insert from UI | Susmitha Palacherla |
| 10/22/2015 | 4.0 | TFS 861 – Allow Warnings for all Modules | Susmitha Palacherla |
| 11/21/2019 | 5.0 | TFS 15803 - changes for warnings workflow in the eCoaching Log application | Susmitha Palacherla |
| 03/23/2020 | 6.0 | TFS 16855 – Add CSR Comments to review | Susmitha Palacherla |

Contents

[1.0 SCR 13479 – Incorporate CSR warnings in eCL 4](#_Toc35932524)

[2.0 SCR 13624 – Inactivate Expired warnings 14](#_Toc35932525)

[3.0 TFS 363/402 Fix issue with duplicate form names for insert from UI 16](#_Toc35932526)

[4.0 TFS 861- Allow Warnings submissions for all Modules 18](#_Toc35932527)

[5.0 TFS 15803 - changes for warnings workflow in the eCoaching Log application 28](#_Toc35932528)

[6.0 TFS 16855 – Capture CSR Comments for Warnings Review 34](#_Toc35932529)

## SCR 13479 – Incorporate CSR warnings in eCL

| Item | Description |
| --- | --- |
| Change Type | New Functionality |
| Change Description | Operations is looking to track Warnings as a new Reason in eCL. |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Tables Updated | Multiple |
| Code doc | Multiple |
| Notes |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test #** | **Action** | **Expected Results** | **Result Pass/Fail** | **Comments** |
|  | Test new Function fn\_strDirectUserHierarchy  SELECT EC].[fn\_strDirectUserHierarchy]  ('susmitha.palacherla','timothy.queen',Getdate()) | Yes | Pass |  |
|  | Test new Function fn\_strDirectUserHierarchy  SELECT [eCoachingDev].[EC].[fn\_strDirectUserHierarchy]  ('susmitha.palacherla','jourdain.augustin', GetDate()) | No | Pass |  |
| 3. | Test new Function fn\_strDirectUserHierarchy  SELECT [EC].[fn\_strDirectUserHierarchy]  ('susmitha.palacherla','timothy.queen','2002-01-01') | No | Pass |  |
| 4. | Test updated procedure sp\_Select\_CoachingReasons\_By\_Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @strModulein = N'CSR',  @strSourcein = N'direct',  @isSplReason = 1,-- 0 for non WARN and CSE  @splReasonPrty = 1, -- use 1 for WARN; 2 for CSE  @strCSRin = N'jourdain.augustin', --  @strSubmitterin = N'timothy.queen'  SELECT 'Return Value' = @return\_value  GO | CoachingReasonID CoachingReason  28 Verbal Warning  29 Written Warning  30 Final Written Warning | Pass |  |
| 5. | Test updated procedure sp\_Select\_CoachingReasons\_By\_Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @strModulein = N'CSR',  @strSourcein = N'indirect',  @isSplReason = 1,-- 0 for non WARN and CSE  @splReasonPrty = 1, -- use 1 for WARN; 2 for CSE  @strCSRin = N'jourdain.augustin', --  @strSubmitterin = N'timothy.queen'  SELECT 'Return Value' = @return\_value  GO | CoachingReasonID CoachingReason | Pass |  |
| 6. | Test updated procedure sp\_Select\_CoachingReasons\_By\_Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @strModulein = N'CSR',  @strSourcein = N'indirect',  @isSplReason = 1,-- 0 for non WARN and CSE  @splReasonPrty = 2, -- use 1 for WARN; 2 for CSE  @strCSRin = N'jourdain.augustin', --  @strSubmitterin = N'timothy.queen'  SELECT 'Return Value' = @return\_value  GO | CoachingReasonID CoachingReason  6 Customer Service Escalation | Pass |  |
| 7. | Test updated procedure sp\_Select\_CoachingReasons\_By\_Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @strModulein = N'CSR',  @strSourcein = N'direct',  @isSplReason = 1,-- 0 for non WARN and CSE  @splReasonPrty = 2, -- use 1 for WARN; 2 for CSE  @strCSRin = N'jourdain.augustin', --  @strSubmitterin = N'timothy.queen'  SELECT 'Return Value' = @return\_value  GO | CoachingReasonID CoachingReason  6 Customer Service Escalation | Pass |  |
| 8. | Test updated procedure sp\_Select\_CoachingReasons\_By\_Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @strModulein = N'CSR',  @strSourcein = N'direct',  @isSplReason = 0,-- 0 for non WARN and CSE  @splReasonPrty = 2, -- use 1 for WARN; 2 for CSE  @strCSRin = N'jourdain.augustin', --  @strSubmitterin = N'timothy.queen'  SELECT 'Return Value' = @return\_value  GO | CoachingReasonID CoachingReason  1 AHT  2 ARC Issue  3 Attendance  4 CCO Processes and Procedures  5 Current Coaching Initiative  7 Feedback  8 HR Guideline Issues  9 OMR / Exceptions  10 Quality  11 Recognition  12 Schedule Adherence  13 Secure Floor Violations | Pass |  |
| 9. | Test updated procedure sp\_Select\_CoachingReasons\_By\_Module  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @strModulein = N'CSR',  @strSourcein = N'indirect',  @isSplReason = 0,-- 0 for non WARN and CSE  @splReasonPrty = 2, -- use 1 for WARN; 2 for CSE  @strCSRin = N'jourdain.augustin', --  @strSubmitterin = N'timothy.queen'  SELECT 'Return Value' = @return\_value | CoachingReasonID CoachingReason  1 AHT  2 ARC Issue  3 Attendance  4 CCO Processes and Procedures  5 Current Coaching Initiative  7 Feedback  8 HR Guideline Issues  9 OMR / Exceptions  10 Quality  11 Recognition  12 Schedule Adherence  13 Secure Floor Violations | Pass |  |
| 10. | Test updated procedure sp\_Select\_SubCoachingReasons\_By\_Reason  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_SubCoachingReasons\_By\_Reason]  @strReasonin = N'Verbal Warning',  @strModulein = N'CSR',  @strSourcein = N'Direct',  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO | SubCoachingReasonID SubCoachingReason  92 Attendance  93 Call Avoidance  94 Conduct  95 Dress Code  48 ETS  96 GDIT Policy  98 Performance  97 Quality  77 Secure Floor Policy  1 Security and Privacy Incident  62 Other: Specify reason under coaching details. | Pass | Retested after adding ETS as a Sub Coaching Source. Failed Initially.  Had to modify sp sp\_Select\_SubCoachingReasons\_By\_Reason to display ETS as a Sub coaching Reason irrespective of Job Code for Warnings related Coaching Reasons.  11 Sub Coaching Reasons returned including ETS as expected after update. |
| 11. | Test updated procedure sp\_Select\_SubCoachingReasons\_By\_Reason  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_SubCoachingReasons\_By\_Reason]  @strReasonin = N'Written Warning',  @strModulein = N'CSR',  @strSourcein = N'Direct',  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO | SubCoachingReasonID SubCoachingReason  92 Attendance  93 Call Avoidance  94 Conduct  95 Dress Code  48 ETS  96 GDIT Policy  98 Performance  97 Quality  77 Secure Floor Policy  1 Security and Privacy Incident  62 Other: Specify reason under coaching details. | Pass | See Comments for # 10 above. |
| 12. | Test updated procedure sp\_Select\_SubCoachingReasons\_By\_Reason  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_SubCoachingReasons\_By\_Reason]  @strReasonin = N'Final Written Warning',  @strModulein = N'CSR',  @strSourcein = N'Direct',  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO | SubCoachingReasonID SubCoachingReason  92 Attendance  93 Call Avoidance  94 Conduct  95 Dress Code  48 ETS  96 GDIT Policy  98 Performance  97 Quality  77 Secure Floor Policy  1 Security and Privacy Incident  62 Other: Specify reason under coaching details. | Pass | See Comments for # 10 above. |
| 13. | Test updated procedure sp\_Select\_SubCoachingReasons\_By\_Reason  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_SubCoachingReasons\_By\_Reason]  @strReasonin = N'Final Written Warning',  @strModulein = N'CSR',  @strSourcein = N'InDirect',  @nvcEmpLanIDin = N'susmitha.palacherla'  SELECT 'Return Value' = @return\_value  GO | SubCoachingReasonID SubCoachingReason | Pass |  |
| 14. | Test Insert from backend  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InsertInto\_Warning\_Log]  @nvcFormName = N'jourdain.augustin-1',  @nvcProgramName = N'NA',  @nvcEmpLanID = N'jourdain.augustin',  @SiteID = NULL,  @nvcSubmitter = N'timothy.queen',  @dtmEventDate = N'2014-10-10',  @intCoachReasonID1 = 28,  @nvcSubCoachReasonID1 = N'92',  @nvcDescription = N'This is a test to insert a warning record.',  @nvcCoachingNotes = N'This is a test to insert a warning record.',  @dtmSubmittedDate = N'2014-10-10',  @ModuleID = 1  SELECT 'Return Value' = @return\_value  GO | 1 record inserted to warning\_log table with identity 1  And 2 records inserted to warning\_log\_reason table | Pass | Obsolete as of 10/22.  Replaced with test # 20 |
| 15. | exec ec.sp\_openkeys  SELECT [WarningID]  ,[FormName]  ,[ProgramName]  ,[SourceID]  ,[StatusID]  ,[SiteID]  ,[EmpLanID]  ,[EmpID]  ,[SubmitterID]  ,[SupID]  ,[MgrID]  ,[WarningGivenDate]  ,[Description]  ,[EC].fn\_Decrypt([Description])  ,[CoachingNotes]  ,[EC].fn\_Decrypt([CoachingNotes])  ,[SubmittedDate]  ,[ModuleID]  ,[Active]  FROM [eCoachingDev].[EC].[Warning\_Log]  GO  CLOSE SYMMETRIC KEY WarnDescKey; | Expecting to see decrypted values for Decsription and CoachingNotes as submitted. | Pass | Obsolete as of 10/22 |
| 16. | USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectReviewFrom\_Warning\_Log]  @strFormIDin = N'Endalkachew.Negussie-1'  SELECT 'Return Value' = @return\_value  GO | Returns the warning log | Pass |  |
| 17. | USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectReviewFrom\_Warning\_Log\_Reasons]  @strFormIDin =N'Endalkachew.Negussie-1'  SELECT 'Return Value' = @return\_value  GO | Returns the warnings logs for the above record in test 16. | Pass |  |
| 18. | USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Warning\_Log\_SUPCSRCompleted]  @strCSRSUPin = N'susmitha.palacherla',  @strSDatein = N'2014-10-01',  @strEDatein = N'2014-10-14',  @bitActive = 1  SELECT 'Return Value' = @return\_value  GO | Returns 2 records where sup is Susmitha palacherla | Pass |  |
| 19. | USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Warning\_Log\_MGRCSRCompleted]  @strCSRMGRin = N'Andy.Ma',  @strSDatein = N'2014-10-01',  @strEDatein = N'2014-10-13',  @bitActive = N'1'  SELECT 'Return Value' = @return\_value  GO | Returns 2 records where mgr is Andy ma | Pass |  |
| 20. | Test Insert from backend  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InsertInto\_Warning\_Log]  @nvcFormName = N'jourdain.augustin-9',  @nvcProgramName = N'NA',  @nvcEmpLanID = N'jourdain.augustin',  @SiteID = NULL,  @nvcSubmitter = N'timothy.queen',  @dtmEventDate = N'2014-10-22',  @intCoachReasonID1 = 29,  @nvcSubCoachReasonID1 = N'92',  @dtmSubmittedDate = N'2014-10-22',  @ModuleID = 1  SELECT 'Return Value' = @return\_value  GO | 1 record inserted to warning\_log table with identity 1  And 1 records inserted to warning\_log\_reason table | Pass |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## SCR 13624 – Inactivate Expired warnings

| Item | Description |
| --- | --- |
| Change Type | New Functionality |
| Change Description | Inactivate all Warning logs 13 weeks after Warning Given Date |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Tables Updated | EC.Warning\_Log |
| Code doc | CCO\_eCoaching\_Warning\_Log\_Create.sql |
| Notes | Using 91 days for 13 weeks. The Inactivation is done using a stored procedure that is called as a step in the Employee Hierarchy load process that is a daily process. For the unit testing the stored procedure will be executed directly in ssms. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test #** | **Action** | **Expected Results** | **Result Pass/Fail** | **Comments** |
|  | Set up a test record that has a warninggivendate older than 91 days.  update ec.warning\_log  set WarningGivenDate = '2014-06-22 00:00:00.000'  where formname = 'eCL-Jourdain.Augustin-739834'  Then Exceute sp [EC].[sp\_InactivateExpiredWarningLogs]  USE [eCoachingDev]  GO  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_InactivateExpiredWarningLogs]  SELECT 'Return Value' = @return\_value  GO | When querying the warning\_log table, of the 4 logs present in the table , 1 log should have Active = 0 and the other 3 Active = 1 | Pass |  |
|  |  |  |  |  |

## TFS 363/402 Fix issue with duplicate form names for insert from UI

| Item | | Description | | | |
| --- | --- | --- | --- | --- | --- |
| Change Type | | Problem Fix | | | |
| Change Description | | Once in a while, users are not able to access "My Dashboard" and/or "My Submissions" and/or "Historical Dashboard", error page is displayed for the users (A system error has occurred. Please contact the system admin for assistance.".  Under the scene, this is the exception caught in debug mode:  Subquery returned more than 1 value. This is not permitted when the subquery follows =, !=, <, <=, >, >= or when the subquery is used as an expression.  The above exception is thrown when there are more than 1 record with the form name passed to sp\_SelectReviewFrom\_Coaching\_Log\_Reasons/sp\_SelectReviewFrom\_Warning\_Log\_Reasons.  Fix: UI doesn't generate the random digits as part of the form name, instead stored procedure will insert the record, get the inserted record primary key (CoachingID/WarningID), append to the form name, and update the inserted record; | | | |
| Test Environment | | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 | | | |
| Code Modules created/updated | | SP [EC].[sp\_InsertInto\_Warning\_Log] | | | |
| Code doc | | CCO\_eCoaching\_Warning\_Log\_Create.sql | | | |
| Notes | |  | | | |
| TEST# | ACTION | | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3.1 | USE [eCoachingDev]  GO  DECLARE @return\_value int,  @isDup bit,  @nvcNewFormName nvarchar(50)  EXEC @return\_value = [EC].[sp\_InsertInto\_Warning\_Log]  @nvcFormName = N'Jourdain.Augustin',  @nvcProgramName = N'Marketplace',  @nvcEmpLanID = N'lili.huang',  @SiteID = 19,  @nvcSubmitter = N'Susmitha.Palacherla',  @dtmEventDate = N'2015-07-20 00:00:00.000',  @intCoachReasonID1 = 28,  @nvcSubCoachReasonID1 = N'91',  @dtmSubmittedDate = N'2015-07-23 15:46:00.000',  @ModuleID = 1,  @isDup = @isDup OUTPUT,  @nvcNewFormName = @nvcNewFormName OUTPUT    SELECT @isDup as N'isDup'  SELECT @nvcNewFormName as N'NewFormName'  SELECT 'Return Value' = @return\_value  GO | Record should be inserted with FormName eCL-Jourdain.Augustin-5463 as WarningID for record is based on identity which is 5463. | P |  |
| 3.2 | Re-Insert above record with same values. | Record should not be inserted as it is a duplicate. | P |  |

## TFS 861- Allow Warnings submissions for all Modules

| Item | Description |
| --- | --- |
| Change Type | Change |
| Change Description | Allow Warnings submissions for all Modules (CSR and Sup already have them) Opening up for LSA, Training and Quality |
| Test Environment | eCoaching\_Dev database on vrivfssdbt02\scord01,1437 |
| Code Modules created/updated | [EC].[Coaching\_Reason\_Selection] Table  [EC].[Email\_Notifications] Table  PROCEDURE [EC].[sp\_Select\_Modules\_By\_Job\_Code]  PROCEDURE [EC].[sp\_InsertInto\_Warning\_Log] |
| Code doc | CCO\_eCoaching\_Log\_Create.sql  CCO\_Warning\_Log\_Create.sql  CCO\_eCoaching\_Dimension\_Table\_Data.xlsx |
| Notes | SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[isCSR]  ,[isSupervisor]  ,[isQuality]  ,[isLSA]  ,[isTraining]  FROM [eCoachingTest].[EC].[Employee\_Selection]  where isQuality = 1  GO  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[CSR]  ,[Supervisor]  ,[Quality]  ,[LSA]  ,[Training]  FROM [eCoachingTest].[EC].[Module\_Submission]  where Quality = 1  GO |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 4.1 | Verify the new column Behavior in Warning\_log table and that column ProgramName does not have Not NULL constraint. | ProgramName (nvarchar(50) NULL)  Behavior (nvarchar(300 NULl0 | P |  |
| 4.2 | Select \* FROM [EC].[Coaching\_Reason\_Selection]  where [CoachingReasonID] in (28,29,30) | 18 rows returned with Quality, LSA and Training columns all set to 1 | P |  |
| 4.3 | Select \* FROM [EC].[Email\_Notifications]  where SubSource = 'Warning' | Verify that a row is returned for each Module(s)  With the following attribute values  Module- Module Name  Submission – UI  Source – Direct  Sub source – Warning  isCSE – 0  Status – Completed  Recipient – NA  Subject – NA  Body – NA  isCCRecipient – 0  CCRecipient - NA | P |  |
| 4.4 | Test submission of warning log for each Module  Quality:  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[isCSR]  ,[isSupervisor]  ,[isQuality]  ,[isLSA]  ,[isTraining]  FROM [eCoachingTest].[EC].[Employee\_Selection]  where isQuality = 1  GO  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[CSR]  ,[Supervisor]  ,[Quality]  ,[LSA]  ,[Training]  FROM [eCoachingTest].[EC].[Module\_Submission]  where Quality = 1  GO  SELECT [Emp\_ID]  ,[Emp\_Name]  ,[Emp\_Email]  ,[Emp\_LanID]  ,[Emp\_Site]  ,[Emp\_Job\_Code]  ,[Emp\_Job\_Description]  ,[Emp\_Program]  ,[Sup\_ID]  ,[Sup\_Name]  ,[Sup\_Email]  ,[Sup\_LanID]  ,[Sup\_Job\_Code]  ,[Sup\_Job\_Description]  ,[Mgr\_ID]  ,[Mgr\_Name]  ,[Mgr\_Email]  ,[Mgr\_LanID]  ,[Mgr\_Job\_Code]  ,[Mgr\_Job\_Description]  ,[Start\_Date]  ,[End\_Date]  ,[Active]  FROM [eCoachingTest].[EC].[Employee\_Hierarchy]  --where sup\_job\_code = 'WSQA50' OR mgr\_job\_code = 'WSQA50'  where Emp\_Job\_Code in ('WACQ02' ,'WACQ12', 'WISO13', 'WACQ03')  GO  Update [EC].[Employee\_Hierarchy]  set Emp\_Job\_Code = 'WACQ13'  where Emp\_ID = '345712'  --318624 --Polimeni, Christine A --Christine.Polimeni  Update [EC].[Employee\_Hierarchy]  set Sup\_ID = '345712'  ,Sup\_Name ='Palacheral, Susmitha C'  ,Sup\_LanID = 'susmitha.palacherla'  where Emp\_ID = '217621'  Update [EC].[Employee\_Hierarchy]  set Sup\_ID = '318624'  ,Sup\_Name ='Polimeni, Christine A'  ,Sup\_LanID = 'Christine.Polimeni'  where Emp\_ID = '217621'  DECLARE @return\_value int,  @isDup bit,  @nvcNewFormName nvarchar(50)  EXEC @return\_value = [EC].[sp\_InsertInto\_Warning\_Log]  @nvcFormName = N'Christine.Polimeni',  @nvcProgramName = N'Marketplace',  @nvcEmpLanID = N'Christine.Polimeni',  @SiteID = 19,  @nvcSubmitter = N'Susmitha.Palacherla',  @dtmEventDate = N'2015-10-22 00:00:00.000',  @intCoachReasonID1 = 28,  @nvcSubCoachReasonID1 = N'91',  @dtmSubmittedDate = N'2015-10-22 15:46:00.000',  @ModuleID = 3,  @nvcbehavior = NULL,  @isDup = @isDup OUTPUT,  @nvcNewFormName = @nvcNewFormName OUTPUT    SELECT @isDup as N'isDup'  SELECT @nvcNewFormName as N'NewFormName'  SELECT 'Return Value' = @return\_value  GO | eCLs should be inserted into the Warning log and Reasons Table on first attempt.  Should receive dup error on second attempt.  For all other employees where Susmitha.palacherla is not the sup or mgr the warning should not be an option in the UI. | P - |  |
| 4.5 | LSA:  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[isCSR]  ,[isSupervisor]  ,[isQuality]  ,[isLSA]  ,[isTraining]  FROM [eCoachingTest].[EC].[Employee\_Selection]  where isLSA = 1  GO  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[CSR]  ,[Supervisor]  ,[Quality]  ,[LSA]  ,[Training]  FROM [eCoachingTest].[EC].[Module\_Submission]  where [LSA] = 1  GO  SELECT [Emp\_ID]  ,[Emp\_Name]  ,[Emp\_Email]  ,[Emp\_LanID]  ,[Emp\_Site]  ,[Emp\_Job\_Code]  ,[Emp\_Job\_Description]  ,[Emp\_Program]  ,[Sup\_ID]  ,[Sup\_Name]  ,[Sup\_Email]  ,[Sup\_LanID]  ,[Sup\_Job\_Code]  ,[Sup\_Job\_Description]  ,[Mgr\_ID]  ,[Mgr\_Name]  ,[Mgr\_Email]  ,[Mgr\_LanID]  ,[Mgr\_Job\_Code]  ,[Mgr\_Job\_Description]  ,[Start\_Date]  ,[End\_Date]  ,[Active]  FROM [eCoachingTest].[EC].[Employee\_Hierarchy]  --where sup\_job\_code = 'WSQA50' OR mgr\_job\_code = 'WSQA50'  where Emp\_Job\_Code in ('WIHD01' ,'WIHD02', 'WIHD03', 'WIHD04')  GO  Update [EC].[Employee\_Hierarchy]  set Emp\_Job\_Code = 'WMPR50'  where Emp\_ID = '345712'  --318624 --Polimeni, Christine A --Christine.Polimeni  Update [EC].[Employee\_Hierarchy]  set Sup\_ID = '345712'  ,Sup\_Name ='Palacheral, Susmitha C'  ,Sup\_LanID = 'susmitha.palacherla'  where Emp\_ID = '220140'  Update [EC].[Employee\_Hierarchy]  set Sup\_ID = '333162'  ,Sup\_Name ='Hamby, Kimberly A'  ,Sup\_LanID = 'Kimberly.Stribling'  where Emp\_ID = '220140'  DECLARE @return\_value int,  @isDup bit,  @nvcNewFormName nvarchar(50)  EXEC @return\_value = [EC].[sp\_InsertInto\_Warning\_Log]  @nvcFormName = N'Brady, Franklin L',  @nvcProgramName = N'Marketplace',  @nvcEmpLanID = N'Brady, Franklin L',  @SiteID = 19,  @nvcSubmitter = N'Susmitha.Palacherla',  @dtmEventDate = N'2015-10-22 00:00:00.000',  @intCoachReasonID1 = 29,  @nvcSubCoachReasonID1 = N'91',  @dtmSubmittedDate = N'2015-10-22 15:46:00.000',  @ModuleID = 4,  @nvcbehavior = NULL,  @isDup = @isDup OUTPUT,  @nvcNewFormName = @nvcNewFormName OUTPUT    SELECT @isDup as N'isDup'  SELECT @nvcNewFormName as N'NewFormName'  SELECT 'Return Value' = @return\_value  GO | eCLs should be inserted into the Warning log and Reasons Table on first attempt.  Should receive dup error on second attempt.  For all other employees where Susmitha.palacherla is not the sup or mgr the warning should not be an option in the UI. | P |  |
| 4.6 | Training:  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[isCSR]  ,[isSupervisor]  ,[isQuality]  ,[isLSA]  ,[isTraining]  FROM [eCoachingTest].[EC].[Employee\_Selection]  where [isTraining] = 1  GO  SELECT [Job\_Code]  ,[Job\_Code\_Description]  ,[CSR]  ,[Supervisor]  ,[Quality]  ,[LSA]  ,[Training]  FROM [eCoachingTest].[EC].[Module\_Submission]  where [Training] = 1  GO  SELECT [Emp\_ID]  ,[Emp\_Name]  ,[Emp\_Email]  ,[Emp\_LanID]  ,[Emp\_Site]  ,[Emp\_Job\_Code]  ,[Emp\_Job\_Description]  ,[Emp\_Program]  ,[Sup\_ID]  ,[Sup\_Name]  ,[Sup\_Email]  ,[Sup\_LanID]  ,[Sup\_Job\_Code]  ,[Sup\_Job\_Description]  ,[Mgr\_ID]  ,[Mgr\_Name]  ,[Mgr\_Email]  ,[Mgr\_LanID]  ,[Mgr\_Job\_Code]  ,[Mgr\_Job\_Description]  ,[Start\_Date]  ,[End\_Date]  ,[Active]  FROM [eCoachingTest].[EC].[Employee\_Hierarchy]  --where sup\_job\_code = 'WSQA50' OR mgr\_job\_code = 'WSQA50'  where Emp\_Job\_Code in ('WTTI02' ,'WTTR12', 'WTTR13', 'WTID13')  and Active = 'A'  GO  Update [EC].[Employee\_Hierarchy]  set Emp\_Job\_Code = 'WTTR40'  where Emp\_ID = '345712'  Update [EC].[Employee\_Hierarchy]  set Sup\_ID = '345712'  ,Sup\_Name ='Palacheral, Susmitha C'  ,Sup\_LanID = 'susmitha.palacherla'  where Emp\_ID = '219324'  DECLARE @return\_value int,  @isDup bit,  @nvcNewFormName nvarchar(50)  EXEC @return\_value = [EC].[sp\_InsertInto\_Warning\_Log]  @nvcFormName = N'MeiJia.Lu',  @nvcProgramName = NULL,  @nvcEmpLanID = N'MeiJia.Lu',  @SiteID = 19,  @nvcSubmitter = N'Susmitha.Palacherla',  @dtmEventDate = N'2015-10-22 00:00:00.000',  @intCoachReasonID1 = 29,  @nvcSubCoachReasonID1 = N'91',  @dtmSubmittedDate = N'2015-10-22 15:46:00.000',  @ModuleID = 4,  @nvcbehavior = N'Production',  @isDup = @isDup OUTPUT,  @nvcNewFormName = @nvcNewFormName OUTPUT    SELECT @isDup as N'isDup'  SELECT @nvcNewFormName as N'NewFormName'  SELECT 'Return Value' = @return\_value  GO | eCLs should be inserted into the Warning log and Reasons Table on first attempt.  Should receive dup error on second attempt.  For all other employees where Susmitha.palacherla is not the sup or mgr the warning should not be an option in the UI. | P |  |

## TFS 15803 - changes for warnings workflow in the eCoaching Log application

| Item | Description |
| --- | --- |
| Change Type | Change |
| Change Description | Warnings submitted go to Pending Employee Review and display in Dashboards per requirements  1 new Reason and new list of sub reasons  Reminders |
| Test Environment | eCoachingDev on f3420-ecldbd01 |
| Code Modules created/updated | [EC].[DIM\_Coaching\_Reason]  [EC].[DIM\_Sub\_Coaching\_Reason]  [EC].[Coaching\_Reason\_Selection]  [EC].[Email\_Notifications]  Several Stored procedures  Reminders\_[Dev][Test[Prod].vbs |
| Code doc |  |
| Notes |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 5.1 | Check for existence of new Coaching Reason  SELECT \* FROM [EC].[DIM\_Coaching\_Reason]  Where CoachingReason = 'Formal Coaching' | 60 Formal Coaching | P |  |
| 5.2 | New SubCoaching Reason List  SELECT \* FROM [EC].[DIM\_Sub\_Coaching\_Reason] | 253- 265 |  |  |
| 5.3 | SELECT \* FROM [EC].[Coaching\_Reason\_Selection]  where CoachingReasonID in (28,29,30,60)  and isActive = 1 | 16\*4 = 64 rows |  |  |
| 5.4 | SELECT \* FROM [EC].[Coaching\_Reason\_Selection]  where CoachingReasonID in (28,29,30,60)  and isActive = 0 | 8\*3=24 rows |  |  |
| 5.5 | DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_CoachingReasons\_By\_Module]  @intModuleIDin = 1,  @strSourcein = N'Direct',  @isSplReason = 1,  @splReasonPrty = 1,  @strEmpIDin = N'233176',  @strSubmitterIDin = N'236464'  SELECT 'Return Value' = @return\_value  GO | CoachingReasonID CoachingReason  60 Formal Coaching  28 Verbal Warning  29 Written Warning  30 Final Written Warning |  |  |
| 5.6 | DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_Select\_SubCoachingReasons\_By\_Reason]  @intReasonIDin = 60,  @intModuleIDin = 1,  @strSourcein = N'Direct',  @nvcEmpIDin = N'236464'  SELECT 'Return Value' = @return\_value  GO | SubCoachingReasonID SubCoachingReason  259 Adherence to schedule  253 Attendance – Hours  255 Attendance – NCNS  254 Attendance – Trends  256 Break Time  257 Call Avoidance  155 Dress Code  260 Falsification  261 Harassment  178 Insubordination  258 Misconduct  263 Performance  264 Quality  95 Security or Privacy Issue  265 Timesheet Violation  262 Other Policy Violation (non-Security/Privacy) |  |  |
| 5.7 | Insert warning log from UI | Inserted with statusid 4 |  |  |
| 5.8 | Verify above submitted log gets included in employees MyPending Count  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Coaching\_Log\_MyPending\_Count]  @nvcUserIdin = N'264599',  @nvcEmpIdin = N'-1',  @nvcSupIdin = N'-1'  SELECT 'Return Value' = @return\_value  GO | 1 |  |  |
| 5.9 | Verify above submitted log shows up in Employees my Pending section  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Coaching\_Log\_MyPending]  @nvcUserIdin = N'264599',  @nvcEmpIdin = N'-1',  @nvcSupIdin = N'-1',  @PageSize = 100,  @startRowIndex = 1,  @sortBy = N'strFormID',  @sortASC = N'y'  SELECT 'Return Value' = @return\_value  GO |  |  |  |
| 5.10 | Verify all pending and completed warning logs show up in sup my Team Warning dashboard count  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Warning\_Log\_MyTeamWarning\_Count\_w]  @nvcUserIdin = N'236464',  @intStatusIdin = -1,  @strSDatein = N'11/1/2019',  @strEDatein = N'11/21/2019'  SELECT 'Return Value' = @return\_value  GO |  |  |  |
| 5.11 | Verify all pending and completed warning logs show up in sup my Team Warning dashboard details  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectFrom\_Warning\_Log\_MyTeamWarning]  @nvcUserIdin = N'236464',  @intStatusIdin = -1,  @strSDatein = N'11/1/2019',  @strEDatein = N'11/21/2019',  @PageSize = 10,  @startRowIndex = 1,  @sortBy = N'strformid',  @sortASC = N'y'  SELECT 'Return Value' = @return\_value  GO |  |  |  |
| 5.12 | Review warning log where coaching reason not ‘Formal Coaching’ and subcoaching reason not in ‘Attendance – Hours’  Attendance – Trends’  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectReviewFrom\_Warning\_Log]  @intLogId = 15499  SELECT 'Return Value' = @return\_value  GO | FC / ATTT and Fc / ATTH are both 0 |  |  |
| 5.13 | Review warning log where coaching reason is ‘Formal Coaching’ and subcoaching reason is ‘Attendance – Hours’  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectReviewFrom\_Warning\_Log]  @intLogId = 15500  SELECT 'Return Value' = @return\_value  GO | FC / ATTH = 1  FC / ATTT = 0 |  |  |
| 5.14 | Review warning log where coaching reason is ‘Formal Coaching’ and subcoaching reason not in ‘Attendance – Hours’  Attendance – Trends’  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectReviewFrom\_Warning\_Log]  @intLogId = 15503  SELECT 'Return Value' = @return\_value  GO | FC / ATTT and Fc / ATTH are both 0 |  |  |
| 5.15 | Verify log selection for reminders for logs that are older than 48 hours if remindercount = 0 or >48 hours from reminder date where remindercount = 1  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectCoaching4Reminder]  SELECT 'Return Value' = @return\_value  GO | 2 logs selected |  |  |
| 5.16 | Verify reminder notification email  Start CoachingReminders sql agent job | Notification sent  Log selection sp does not pick them up until they meet the next reminder selection criteria |  |  |
| 5.17 | Select Notification attributes like Subject, High importance flag and body and recipients |  |  |  |

## TFS 16855 – Capture CSR Comments for Warnings Review

| Item | Description |
| --- | --- |
| Change Type | Change |
| Change Description | Capture CSR Comments during Warning log review. |
| Test Environment | eCoachingDev on f3420-ecldbd01 |
| Code Modules created/updated | sp\_Update\_Review\_Coaching\_Log\_Employee\_Pending  sp\_SelectReviewFrom\_Warning\_Log |
| Code doc | sp\_Update\_Review\_Coaching\_Log\_Employee\_Pending.sql  sp\_SelectReviewFrom\_Warning\_Log.sql |
| Notes |  |

| TEST# | ACTION | EXPECTED RESULTS | RESULTS  P/F/I | COMMENTS |
| --- | --- | --- | --- | --- |
| 6.1 | Review log as CSR  Make yourself as warning log owner and CSR job code and go to the MyDashboard page and click on the log name from the list.  Enter Comments into the Comments text box | Comments saved successfully to the warning\_log table and encrypted on save. |  | 15515 |
| 6.2 | Review warning log  DECLARE @return\_value int  EXEC @return\_value = [EC].[sp\_SelectReviewFrom\_Warning\_Log]  @intLogId = 15515  SELECT 'Return Value' = @return\_value  GO | Decrypted comments displayed. |  |  |