-------------------------Unassigned Request------------------------------------

Feature:Verify Unassigned Request Id feature

*Verify Request id feature which is not assigned to any two approver*

Background:Login to the RMS-Backbone Application

Given User is login to the application as "*Backbone Admin*" role

When User selects "Un-assigned Request" option from Request dropdown

Then User should be able to see the "Unassigned Requests" header in the page

Scenario:Verify Unassigned Requests data table verification

Given "Un-assigned Request" data table is displayed

Then "Request Id" link is displayed and clickable

And User should be able to see below column headers for data table

| *ColumnHeader* |

| Request Id |

| User |

| Region Name |

| Role Name |

| Traning Status |

Scenario: Verify Request id popup open and close functionality

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to close the popup using the "Close" button icon

Scenario:Verify Unassigned Request Id Default Status

Given "Request Id" is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see "Training Not Started" as default status

Scenario:Verify Unassigned Request Id popup- Training progress Bar

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see below labels for "Progress Bar" section

| *StatusName* |

| Request Created |

| Nomination sent To LSC |

| Awaiting Training Assignment |

| Training in Progress |

| Training Complted |

| Two Approver |

Scenario:Verify Unassigned Request Id popup- User information

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see below labels for "User Information" section

| *LabelName* |

| First Name |

| Last Name |

| Title |

| Status |

| NTID |

| User type |

Scenario:Verify Unassigned Request Id popup-Request Details information

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "Request Details" header

And User should be able to see below labels for "Request Details" section

| *LabelName* |

| Region |

| Role |

| RMS Request Date |

| RMS Request Created By |

Scenario:Verify Unassigned Request Id popup-Saviynt Request Details information

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "Saviynt Request Details" header

And User should be able to see below labels for "Saviynt Request Details" section

| *LabelName* |

| Saviynt Role |

| Saviynt Request Id |

| Requestor NTID |

| Request Date |

| First Level Approval Date |

| Request Comments |

#Need to Verify

Scenario: Verify Unassigned Request Id popup-Training Status information

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "Training Details" header

And User should be able to see the "No Trainings Present" message

Scenario: Verify Unassigned Request Id popup-TWO Approver information

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "TWO Approvers" header

And User should be able to see the "No TWO approvers found for this Region and Role" message

Scenario: Verify Unassigned Request Id popup-Requests Action log

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "Request Action Log" header

And User should be able to see below labels for "Action Requests log" section

| *LabelName* |

| Action |

| Action By |

| Date Actioned |

Scenario: Verify Unassigned Request Id popup-Approval Action

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "Approval Action" header

And User should be able to see the "Override Training Process?" label associated with "Yes" and "No" radiobutton

And User should be able to see the "Action In Saviynt?" label associated with "Checkbox" element

And "Check Saviynt Status" link is displayed and clickable

Scenario:Verify Unassigned Request id-Comments input field

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to enter the text in comment input field

Scenario Outline:Verify Unassigned Request id-Mandatory Comments input field

Given "Unassigned Request" popup is displayed

And User should be able to see the "Action In Saviynt?" label associated with "Checkbox" element

When User clicks on the <ButtonName> button

Then Comments input field should be highlighted as a mandatory field

And "Unassigned Request Details" popup should not be closed

Examples:

| *ButtonName* |

| Approve |

| Reject |

Scenario:Verify Unassigned Request id-Approve and Reject button

Given "Request Id" link is displayed and clickable

When User clicks on the "Request Id" link

Then User should be able to see the request details popup

And User should be able to see the "Action In Saviynt?" label associated with "Checkbox" element

And "Approve" button is displayed and clickable

And "Reject" button is displayed and clickable

Scenario:Verify Unassigned Request id- Reject Request

Given "Unassigned Requests" popup is displayed

And User should be able to see the "Action In Saviynt?" label associated with "Checkbox" element

When User enters the comments text and clicks on the "Reject" button

Then "Unassigned Requests" popup should be closed

And Request should not be visible in the "Unassigned Requests" table

#After Reject - Current Status changed - Request Rejected

Scenario:Verify Unassigned Request id- Approve Request

Given "Unassigned Requests" popup is displayed

And User should be able to see the "Action In Saviynt?" label associated with "Checkbox" element

When User enters the comments text and clicks on the "Approve" button

Then "Unassigned Requests" popup should be closed

And Request should not be visible in the "Unassigned Requests" table

#After Approve - Current Status changed - Request Completed

------------------------- Non UMT Request ------------------------------------

Feature: Verify Non UMT Request Id feature

*Verify Request Id feature which is Not available in User management Team*

Background:Login to the RMS-Backbone Application

Given User is login to the application as "*Backbone Admin*" role

When User selects "Non UMT Request" option from Request dropdown

Then User should be able to see the "Non UMT Requests" header in the page

Scenario:Verify Non UMT Request data table verification

Given "Non UMT Request" data table is displayed

Then "Saviynt Req Id" link is displayed and clickable

And User should be able to see below column headers for data table

| *ColumnHeader* |

| Saviynt Req Id |

| Requestor NTID |

| Region |

| Role |

| Saviynt Role |

| Request Date |

Scenario: Verify Saviynt Request id popup open and close functionality

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to close the popup using the "Close" button icon

Scenario:Verify Saviynt Request Id popup-Saviynt Request Details information

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to see the "Saviynt Request Details" header

And User should be able to see below labels for "Saviynt Request Details" section

| *LabelName* |

| Saviynt Role |

| Saviynt Request Id |

| Requestor NTID |

| Request Date |

| First Level Approval Date |

| Request Comments |

Scenario: Verify Saviynt request id popup-Reason information

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to see the "Reason" header

And User should be able to see "Unable to process the request as the user is not found in User Management Team" message

Scenario: Verify Saviynt request id popup-TWO Approver information

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to see the "TWO Approvers" header

And User should be able to see "No Approvers" message

Scenario:Verify Saviynt request id-Comments input field

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to see the "Add Comments" header

And User should be able to see comment input field below the "Add Comments" header

And User should be able to enter the text in comment input field

Scenario Outline:Verify Saviynt request id-Mandatory Comments input field

Given "Missing UMT Request" popup is displayed

When User clicks on the <ButtonName> button

Then Comments input field should be highlighted as a mandatory field

And "Missing UMT Request" popup should not be closed

Examples:

| *ButtonName* |

| Approve |

| Reject |

Scenario:Verify Saviynt request id-Approve and Reject button

Given "Saviynt Request Id" is displayed and clickable

When User clicks on the "Saviynt Request Id" link

Then User should be able to see the request details popup

And "Approve" button is displayed and clickable

And "Reject" button is displayed and clickable

Scenario:Verify Saviynt request id- Reject Request

Given "Missing UMT Request" popup is displayed

When User enters the comments text and clicks on the "Reject" button

Then "Missing UMT Request" popup should be closed

Then Request should not be visible in the "NON UMT Request" table

Scenario:Verify Saviynt request id- Approve Request

Given "Missing UMT Request" popup is displayed

When User enters the comments text and clicks on the "Approve" button

Then "Missing UMT Request" popup should be closed

Then Request should not be visible in the "NON UMT Request" table

And Request should have "Request Created" state

#And Request should have "Request Created" state

------------------------- Non UMT Request ------------------------------------

Feature: Verify Missing Saviynt Request feature

*Verify Requests which could not be inserted into the RMS server because of Region and Role Mismatch*

Background:Login to the RMS-Backbone Application

Given User is login to the application as "*Backbone Admin*" role

When User selects "Missing Saviynt Request" option from Request dropdown

Then User should be able to see the "Missing Saviynt Requests" header in the page

Scenario:Verify Missing Saviynt Request data table verification

Given "Missing Saviynt Requests" data table is displayed

Then "Saviynt Req Id" link is displayed and clickable

And User should be able to see below column headers for data table

| *ColumnHeader* |

| Saviynt Req Id |

| Requestor NTID |

| Region |

| Role |

| Request Date |

Scenario: Verify Request id popup open and close functionality

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to close the popup using the "Close" button icon

Scenario:Verify Saviynt Request Id popup-Saviynt Request Details information

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to see the "Saviynt Request Details" header

And User should be able to see below labels for "Saviynt Request Details" section

| *LabelName* |

| Saviynt Role |

| Saviynt Request Id |

| Requestor NTID |

| Request Date |

| First Level Approval Date |

| Request Comments |

Scenario: Verify Saviynt request id popup-Reason Information

Given "Saviynt Req Id" link is displayed and clickable

When User clicks on the "Saviynt Req Id" link

Then User should be able to see the request details popup

And User should be able to see the "Reason" header

------------------------- Assign Backbone Role ------------------------------------

Feature: Verify Assign Backbone Role feature

Background:Login to the RMS-Backbone Application

Given User is login to the application as "*Backbone Admin*" role

When User selects "Assign Backbone Role" option from Request dropdown

Then User should be able to see the "Backbone Role Details" header in the page

Scenario:Verify Access Role for Users section

Given User should be able to see the "ACCESS ROLE USERS" header

Then User should be able to see the "User Name" label associated with "Textbox" element

And User should be able to see the "Backbone Role" label associated with "Dropdown" element

And User should be able to see the "Region" label associated with "Dropdown" element

And User should be able to see the "Site Role" label associated with "Dropdown" element

And "Search" button is displayed and clickable

And "Add To List" button is displayed and clickable

Scenario:Verify Logged-in Default User Name

Given User should be able to see the "ACCESS ROLE USERS" header

Then User should able to see logged-in user name in textbox

Scenario:Verify Assign Backbone Admin Backbone Role when user have already access

Given User selects "*Backbone Admin*" option from "*Backbone Role*" dropdown

Then "Add To List" button is displayed and clickable

And User should be able to see the "Role already added for this user" message

And User should be able to close the popup using the "OK" button icon

Scenario:Verify Assign Backbone Admin Backbone Role when user don't have access

Given User selects "*Backbone Admin*" option from "*Backbone Role*" dropdown

When User clicks on the "*Add To List*" button

And User should be able to see below column headers for data table

| *ColumnHeader* |

| User |

| Role |

| Region |

| Site |

| Remove |

And user clicks on the "Submit" button

And User should be able to see the "Access provided Successfully" message

And User should be able to close the popup using the "OK" button icon

Scenario:Verify Assign BB Support Backbone Role when user have already access

Given User selects "*BB Support*" option from "*Backbone Role*" dropdown

Then "Add To List" button is displayed and clickable

And User should be able to see the "Role already added for this user" message

And User should be able to close the popup using the "OK" button icon

Scenario: Verify Assign BB Support Backbone Role when user don't have access

Given User selects "*BB Support*" option from "*Backbone Role*" dropdown

When User clicks on the "*Add To List*" button

And User should be able to see below column headers for data table

| *ColumnHeader* |

| User |

| Role |

| Region |

| Site |

| Remove |

When User clicks on the "*Submit*" button

And User should be able to see the "Access provided Successfully" message

And User should be able to close the popup using the "OK" button icon

Scenario:Verify Assign TWO Approver Backbone Role when user have already access

Given User selects "*TWO Approver*" option from "*Backbone Role*" dropdown

When User selects "*AGT*" option from "*Region*" dropdown

And User selects "*CRP Reviewer*" option from "*Site Role*" dropdown

When User clicks on the "*Add To List*" button

And User should be able to see the "Role already added for this user" message

And User should be able to close the popup using the "OK" button icon

Scenario: Verify Assign TWO Approver Backbone Role when user don't have access

Given User selects "*TWO Approver*" option from "*Backbone Role*" dropdown

When User selects "*AGT*" option from "*Region*" dropdown

And User selects "*CRP Reviewer*" option from "*Site Role*" dropdown

When User clicks on the "*Add To List*" button

And User should be able to see below column headers for data table

| *ColumnHeader* |

| User |

| Role |

| Region |

| Site |

| Remove |

Then User clicks on the "Submit" button

And User should be able to see the "Access provided Successfully" message

And User should be able to close the popup using the "OK" button icon

Scenario:Verify Role Assign Details section

Given User should be able to see the "ROLE ASSIGN DETAILS" header

And User should be able to see the "Backbone Role" label associated with "Dropdown" element

And User should be able to see the "Region" label associated with "Dropdown" element

And User should be able to see the "Site Role" label associated with "Dropdown" element

And "*Clear Filter*" button is displayed and clickable

And User should be able to see below column headers for data table

| *ColumnHeader* |

| User |

| Backbone Role |

| Region |

| SiteRole Name |

| Remove Access |

And "ROLE ASSIGN DETAILS" data table is displayed

Scenario Outline: Verify Clear filter functionality

Given User should be able to see the "ROLE ASSIGN DETAILS" header

And User selects "*TWO Approver*" option from "*Backbone Role*" dropdown

When User selects "*AGT*" option from "*Region*" dropdown

And User selects "*CRP Reviewer*" option from "*Site Role*" dropdown

And User should be able to see filtered data in the "ROLE ASSIGN DETAILS" table

Then User clicks on the "Clear filter" button

Then User should be able to see unfiltered data in the "ROLE ASSIGN DETAILS" table

Scenario Outline: Verify Remove Assign Access Role

Given User should be able to see the "ROLE ASSIGN DETAILS" header

And "ROLE ASSIGN DETAILS" data table is displayed

When User clicks on remove access using cross icon

And User should be able to see the "Do you want to remove the access?" message

Then User clicks on the "Yes" button

And User should be able to see the "Access removed successfully" message

And User should be able to close the popup using the "OK" button icon

Then Role should not be visible in the "ROLE ASSIGN DETAILS" table

Scenario Outline: Verify when user don't want to Remove Access Role

Given User should be able to see the "ROLE ASSIGN DETAILS" header

And "ROLE ASSIGN DETAILS" data table is displayed

When User clicks on remove access using cross icon

And User should be able to see the "Do you want to remove the access?" message

Then User clicks on the "No" button

Then "Action" popup should be closed

------------------------- AboutUser ------------------------------------

Feature: Verify AboutUser

*User information is display in About user page*

Background:Login to the RMS-Backbone Application

Given User is login to the application as "*Backbone Admin*" role

When User clicks on the "*About user*" header tab

Then User should be able to see the "User Profile" header in the page

Scenario: Verify User Profile details

Given User profile details is displayed on the page

Then User should be able to see the full name in the user profile details section

And User should be able to see the role in the user profile details section

Then User should be able to see the user department in the user profile details section

Then User should be able to see the user email in the user profile details section

Scenario: Verify User Status

Given User status details is displayed on the page

Then User should be able to see the below labels for "User Status" section

| *LabelName* |

| BU |

| Status |

| Company |

Scenario: Verify User Progress Menu Functionality

Given User should able to see the progress menu is displayed on the page

Then User should able to see below tabs for "Progress Menu" section

| *MenuName* |

| Current Access |

| Cerification |

| In progress |

| Closed Request |

Scenario:Verify progress menu Certification option table headers

Given User should able to see the progress menu is displayed on the page

When User selects "Certification" option from Progress Menu

Then User should be able to see below column headers for data table

| *ColumnHeader* |

| Certification Title |

| Completed On |

| Certification Status |

Scenario:Verify progress menu Current Access option table headers

Given User should able to see the progress menu is displayed on the page

When User selects "Current Access" option from Progress Menu

Then User should be able to see below column headers for data table

| *ColumnHeader* |

| Request Id |

| Region |

| Role |

| Access Requested On |

Scenario:Verify progress menu In Progress option table headers

Given User should able to see the progress menu is displayed on the page

When User selects "In Progress" option from Progress Menu

Then User should be able to see below column headers for data table

| *ColumnHeader* |

| Request Id |

| Region |

| Role |

| Access Requested On |

Scenario:Verify progress menu Closed Requests option table headers

Given User should able to see the progress menu is displayed on the page

When User selects "Closed Requests" option from Progress Menu

Then User should be able to see below column headers for data table

| *ColumnHeader* |

| Request Id |

| Region |

| Site |

| Access Granted On |

------------------------- Dashboard ------------------------------ --

Feature: Dashboard

*User can Export Two Approver or Backbone admin data into Excel sheet*

Background:Logged-in to the RMS-Backbone Application

Given User is login to the application as "*Backbone Admin*" role

When User clicks on the "*Dashboards*" header tab

Then User should be able to see the "*Dashboard*" page

Scenario:Verify Region dropdown functionality

Given "*Region*" dropdown is displayed and enabled on the page

When User selects "*AGT*" option from "*Region*" dropdown

Then "*AGT*" option should be selected in "*Region*" dropdown

And User should be able to see "Select a Region and Role to see details in a table" message

And User should be able to see "Click "Export all" to download the list "Two Approver's and "Backbone Role" for AGT region" message

Scenario:Verify Role dropdown functionality

Given "*Role*" dropdown is displayed and enabled on the page

When User selects "*Business Approver*" option from "*Role*" dropdown

Then "*Business Approver*" option should be selected in "*Role*" dropdown

Scenario:Verify Search functionality

Given "Region" dropdown and "Role" dropdown is displayed and enabled on the page

When User selects "*AGT*" option from "*Region*" dropdown

And User selects "*PM Operations Manager*" option from "*Role*" dropdown

And "Search" button is displayed and clickable

Then Result should be displayed on the table

Scenario: Verify Multiple Role selection

Given "*Role*" dropdown is displayed and enabled on the page

When User selects multiple options from "*Role*" dropdown

| *OptionName* |

| PM Operations Manager |

| PM Display & Reporting |

| Business Approver |

Then User should be able to see selected options

Scenario:Verify Multiple Role deselection

Given User has selected multiple options "*Role*" dropdown

| *OptionName* |

| PM Operations Manager |

| PM Display & Reporting |

| Business Approver |

When User deselects multiple options from "*Role*" dropdown

| *OptionName* |

| PM Operations Manager |

| PM Display & Reporting |

Then User should not be able to see deselected options

Scenario:Verify Export All button Functionality

Given User selects "*AGT*" option from "*Region*" dropdown

And "*Export All*" button is displayed and clickable

When User clicks on the "*Export All*" button

Then File of "*.xls*" type should sheet downloaded successfully

Scenario:Verify TWO APPROVER tab

Given User should be able to see "*TWO APPROVER*" tab

And "*TWO APPROVER*" tab is selected

Then User should be able to see below column headers for data table

| *ColumntHeader* |

| User |

| Authority |

| Region |

| Site Role |

Scenario:Verify Backbone Role tab

Given User should be able to see "*BACKBONE ROLE*" tab

And "*BACKBONE ROLE*" tab is selected

Then User should be able to see below column headers for data table

| *ColumntHeader* |

| User |

| Authority |

| Region |

| Site Role |

Scenario: Verify Export To Excel -TWO Approver tab Functionality

Given User should be able to see "*TWO APPROVER*" tab

And "*TWO APPROVER*" tab is selected

And "*Export To Excel*" button is displayed and clickable

When User clicks on the "*Export To Excel*" button

Then File of "*.xls*" type should sheet downloaded successfully

Scenario: Verify Export To Excel -Backbone Role tab Functionality

Given User should be able to see "*BACKBONE ROLE*" tab

And "*BACKBONE ROLE*" tab is selected

And "*Export To Excel*" button is displayed and clickable

When User clicks on the "*Export To Excel*" button

Then File of "*.xls*" type should sheet downloaded successfully

------------------------ Login------------------------------------

Feature: Verify Login Functionality

Background: Navigation to the RMS-Backbone Application

Given User navigate to the RMS-Backbone application

Scenario Outline: Login with valid Credentials

When User enter the valid username <UserName>

And User enter the valid password <Password>

And User clicks on the "*Sign in*" button

Then User should be able to see the "Requests Details" header in the page

Examples:

| *UserName* | *Password* |

| prajakta.ubarhande@bp.com | Prajakta@1995 |

| abc@bp.com | ABC@123 |

Scenario Outline: Login with Invalid UserName

When User enter the invalid username <UserName>

And User enter the valid password <Password>

And User clicks on the "*Sign in*" button

Then User should not be able to see the "Requests Details" header in the page

Examples:

| *UserName* | *Password* |

| prajakta@bp.com | Prajakta@1995 |

Scenario Outline: Login with Invalid Password

Then User enter the valid username <UserName>

And User enter the invalid password <Password>

And User clicks on the "Sign in" button

Then User should not be able to see the "Requests Details" header in the page

Examples:

| *UserName* | *Password* |

| prajakta.ubarhande@bp.com | Prajakta@19 |

Scenario Outline: Login with Invalid UserName and Password

Then User enter the invalid username <UserName>

And User enter the invalid password <Password>

And User clicks on the "Sign in" button

Then User should not be able to see the "Requests Details" header in the page

Examples:

| *UserName* | *Password* |

| prajakta.ubarhande@bp.com | Prajakta@19 |