Scenario Description:

Update Item Selection Algorithm from UI

Version Control

Version #	Date	Author	Description
1.0	03/06/2014	MR	Initial Draft

Test Scripts:

The following scripts, links and pages consist of updating and validating Item Selection Algorithm from Test Authoring UI

Test Components:

This test scenario covers the following high-level test requirements (see scripts below for specific requirements covered by each test script):

- **1.0 Item Selection Algorithm**

Requirements Traceability:

RADTIB 5, RADTIB 5.1, RADTIB 5.2, RADTIB 5.3, RADTIB 5.4, RADTIB 6, RADTIB 6.1, RADTIB 6.2, RADTIB 6.3, RADTIB 6.4, RADTIB 6.5, RADTIB 6.6, RADTIB 6.7, RADTIB 6.8, RADTIB 22.1, RADTIB 22.2, RADTIB 22.3, RADTIB 22.4, RADTIB 22.5

Testing Requirements:

- **Create a version of the item selection algorithm that is provided as major.minor version string. It is acceptable to have multiple item selection algorithms with the same name so long as the versions are different. This is done to accommodate new versions of a item selection algorithm and maintain compatibility with older versions for an existing test.**
- **Item selection algorithms have parameters, but this table only indicates the number of parameters Create The following parts of an item selection rule parameter.**
- **Item selection algorithms shall be configurable in Test Authoring so that new item selection algorithms and their parameters can be added at any time.**
- **Configured versions of item selection algorithms shall have a major.minor version number.**
- **The Test Segment user interface shall present configured item selection algorithms for selection by the Test Author, and shall provide a dynamic user interface for entering configuration parameters specific to the selected algorithm.**

- **Test Authoring shall provide a setup function that analyzes the external inputs for configuration of item selection algorithms and provide users with appropriate error messages if something is wrong with the external configuration.**
- **The system will support a minimum of three item selection algorithms: fixedform, field test and adaptive**

Script #:1.1 Item Selection Algorithm

Purpose:

Update and validate Item Selection Algorithm

Assumptions:

- Test Spec Bank, ProgMan, Permission, Test Item Bank, Core Standard systems are up and running

Prerequisites:

- Test Author application is online and user has valid credentials to login
- At least one Item Selection Algorithm is available that has not been used in any tests

Test case Steps

Step #	Test Action	Expected Results	Pass/ Fail
Login			
L1.0	Open a supported browser and access link http://drcamp-dev.opentestsystem.org:8080/	SSO login page displays	
L1.1	Enter valid user ID and password	TEST AUTHOR Dashboard page displays	
Item Selection Algorithm			
ISA1.0	Click Item Selection Algorithms from Settings wheel	Item Selection Algorithm Search page displays with list of Item Selection Algorithm in the search results grid	
ISA1.1	Click Edit icon from the Item Selection Algorithm row	Item Selection Algorithm view page displays	
ISA1.2	Click Edit button	Item Selection Algorithm edit page displays	

Step #	Test Action	Expected Results	Pass/ Fail
ISA1.3	Update any values Update parameter details Click Save	Updates are saved Item Selection Algorithm view page displays	
ISA1.4	Verify that Item Selection Algorithm view page displays all the changes	Item Selection Algorithm view page displays all the changes	
ISA1.5	Click Item Selection Algorithm from left navigation	Item Selection Algorithm Search page displays with updated Item Selection Algorithm	
Database			
D1.0	Logon to Mongo database collection Execute query to verify that Item Selection Algorithm has been updated	Item Selection Algorithm has been updated	

Test Execution

Date/Time	Tester	Test ID	Test Phase	Status
12/09/2013	RM, SM, SK, MR	UI	1	Pass