Test Plan

Project Name: Digital Dash

Reference Document: Your SRS doc

Created by: Kenny's Krew

Date of creation: 12/12/16

Date of review: 12/15/16

Login & Logout Test Cases

Test Case ID: TC_LG_01 (References SRS, Section 2.1)

Objective: Successful login to system

Preconditions: Viewing the Login page, user has a valid account with the system. Assume some user account data has been established.

Step Number	Step Description	Test Data	Expected Result
1.	Enter username field	An Active Directory username	User is authenticated, logged into system, and given access to the digital dash interface.
2.	Enter password field	A password to above Active Directory account	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	User presses "login" on client side	A request to login via active directory from the client side with a username, password, and an environment.	

Postconditions: User is logged in, the event is written to the log.

Test Case ID: TC_LG_02 (References SRS, Section 2.1)

Objective: Error message on unsuccessful login to system: "incorrect username or password"

Preconditions: Viewing the Login page.

Step Number	Step Description	Test Data	Expected Result
1.	Enter username	Empty username	Error message: "Missing: Username & Password"
2.	Enter password	Empty password	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	Click "Login" button	A request to login via active directory from the client with an environment.	

Postconditions: User is not logged in.

Test Case ID: TC_LG_03 (References SRS, Section 2.1)

Objective: Error message on unsuccessful login to system: invalid password

Preconditions: Viewing the Login page.

Step Number	Step Description	Test Data	Expected Result
1.	enter username	A valid username	Error message: "incorrect username or password"
2.	enter password	An invalid password	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	click "Login" button	A request to login via active directory from the client with a valid username, an invalid password, and an environment.	

Postconditions: User is not logged in.

Test Case ID: TC_LG_04 (References SRS, Section 2.1)

Objective: Error message on unsuccessful login to system: invalid username

Preconditions: Viewing the Login page.

Step Number	Step Description	Test Data	Expected Result
1.	enter username	A invalid user name	Error message: "incorrect username or password"
2.	enter password	An valid password	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	click "Login" button	A request to login via active directory from the client with an invalid username, a valid password, and an environment.	

Postconditions: User is not logged in.

Test Case ID: TC_LG_05 (References SRS, Section 2.1)

Objective: Error message on unsuccessful login to system: missing credential username

Preconditions: Viewing the Login page.

Step Number	Step Description	Test Data	Expected Result
1.	enter username	Empty username	Error message: "Missing: Username".
2.	enter password	A valid password	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	click "Login" button	A request to login via active directory from the client with an empty username, a valid password, and an environment.	

Postconditions: User is not logged in.

Test Case ID: TC_LG_06 (References SRS, Section 2.1)

Objective: Error message on unsuccessful login to system: missing credential password.

Preconditions: Viewing the Login page.

Step Number	Step Description	Test Data	Expected Result
1.	enter username	A valid user name	Error message: "Missing: Password".
2.	enter password	Empty password	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	click "Login" button	A request to login via active directory from the client with a valid username, an empty password, and an environment.	

Postconditions: User is not logged in.

Test Case ID: TC_LG_07 (References SRS, Section 2.1)

Objective: Error message on unsuccessful login to system: invalid username and password

Preconditions: Viewing the Login page.

Step Number	Step Description	Test Data	Expected Result
1.	enter username	A invalid user name	Error message: invalid username and password.
2.	enter password	An invalid password	
3.	Choose Azure Active Directory Environment (Development, Test, QA, and Production)	A selected environment from the login menu.	
4.	click "Login" button		

Postconditions: User is not logged in.

Test Case ID: TC_LGT_01 (References SRS, Section 2.7)

Objective: If the user is logged in to the system, they will successfully log out.

Preconditions: User must be logged in.

Step Number	Step Description	Test Data	Expected Result
1.	clicks "Logout"		User is directed to the homepage (login screen) and remove user credentials for the session

Postconditions: User logged out of the application

Test Case ID: TC_LGT_02 (References SRS, Section 2.7)

Objective: If the user is not logged in to the system, they will notified of their current status (logged out).

Preconditions: User must not be logged in.

Step Number	Step Description	Test Data	Expected Result
1.	clicks "Logout"		System displays an error message, "not currently logged in".

Postconditions: User remains logged out of the application

Peer Review Test Cases

Test Case ID: TC_View_PR_01 (References SRS, Section 2.6)

Objective: View current Peer Review request

Preconditions: User is already logged in

Step Number	Step Description	Test Data	Expected Result
1.	Clicked PeerReview on the top Navigation bar		Show the dropdown menu of current Peer Review request.

Postconditions: Showed Peer Review dropdown

Test Case ID: TC_PR_01 (References SRS, Section 2.6)

Objective: Admin approves pending macro to be run

Preconditions: User logged in as Admin

Step Number	Step Description	Test Data	Expected Result
1.	Admin selects peer review entry from top right menu drop-down (labeled peer review)	PeerReview.db	Pending Macro is displayed.
2.	Admin clicks "Accept" for process		Pending process has status changed to be run. Message is displayed that action has been taken.

Postcondition: Process is accepted to be run. Journal entry is written to log file with specified data: "Review: Approval, Admin Name, AdminID, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" . Admin and Initiator receive a notification that the macro request has been approved via email.

Test Case ID: TC_PR_02 (References SRS, Section 2.6)

Objective: Admin rejects pending macro to be run

Preconditions: User logged in as Admin

Step Number	Step Description	Test Data	Expected Result
1.	Admin selects peer review entry from top right menu	PeerReview.db	Pending Macros are displayed.

	drop-down (labeled peer review)		
2.	Admin clicks "decline" for process		Pending process is removed from PeerReview database. Message is displayed that action has been taken.

Postcondition: Process is declined to be run. Journal entry is written to log file with the specified data: "Review: Denial, Admin Name, AdminID, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date". Admin and Initiator receive a notification that the macro request has been declined via email.

Log Test Cases

Test Case ID: TC_VL_01 (References SRS, Section 2.5)

Objective: View contents of the log on the webpage.

Preconditions: User is logged in.

Step Number	Step Description	Test Data	Expected Result
1.	click "Logs" button	log.txt	Log contents are output to front-end.

Postconditions: Contents of 'log.txt' are displayed.

Test Case ID: TC_WL_01

Objective: Check that the log is written to after successful execution and addToLog is instantiated.

Preconditions: Macro fields are correctly filled in and send for processing.

Step Number	Step Description	Test Data	Expected Result
1.		formatted log string	Log contents are modified to include previous logs and newly added format string.

Postconditions: Contents of 'log.txt' are updated and displayed.

View Homepage Test Cases

Test Case ID: TC_View_Homepage_01

Objective: View the homepage

Preconditions: User is already logged in

Step Number	Step Description	Test Data	Expected Result
1.	Clicked DigitalDash on the top left side of the Navigation bar		Navigate to homepage if currently not on the homepage, otherwise do nothing.

Postconditions: On Index Page.

Test Case ID: TC_View_Homepage_02

Objective: View Index Page

Preconditions: User is already logged in

Step Number	Step Description	Test Data	Expected Result
1.	Clicked DigitalDash on the top left side of the Navigation bar		Navigate to Index Page if currently not on Index Page, otherwise do nothing.

Postconditions: On Index Page.

Update Macro Test Cases

Test Case ID: TC_View_Update_01 (References SRS, Section 2.2)

Objective: View specific update macro page

Preconditions: User is already logged in

Step Number	Step Description	Test Data	Expected Result
1.	Clicked update in the body of the page		Show the dropdown menu of update macro options

2.	Clicked specific update macro option		Show the selected update macro page

Postconditions: on selected update macro page

Test Case ID: TC_Update_Macro_01 (References SRS, Section 2.2)

Objective: Update specific macro with required parameters

Preconditions: User logged in system and on specific update macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Selected required parameters from drop-down	LibertyMutual.db Input: Fields are filled in correctly and follow the specified regex	If this is first field, it will update available options in second field.
2.	User Fills in the update data		
3.	User selects Exec to execute the macro		User is prompted "Are you sure?"
4.	User selects "Yes"		The update macro information is packaged into an update object and logged into the peer review. User receives alert box saying request has been processed.

Postconditions: Macro is pending for approval in the peer review database. "Peer Review, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" is appended to the end of 'log.txt'.

Test Case ID: TC_Update_Macro_02 (References SRS, Section 2.2)

Objective: Execute a macro by bypassing peer review

Preconditions: User with specified credentials is logged in and is on specific update macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select and update parameters of macro	LibertyMutual.db Input: Fields are filled in correctly and follow the specified regex	A macro has been selected to update from the database with specified parameters
2.	User selects "Urgent Execute"		User is prompted "Are you sure?"
3.	User selects "Yes"		The user credentials are verified by the server. The updated macro information is packaged into an update object and written to the driver tables. User receives alert box indicating the deletion of the macro.

Postconditions: The specified macro has been appended to the driver tables. "Override, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" is appended to the end of 'log.txt'.

Test Case ID: TC_Update_Macro_03 (References SRS, Section 2.2)

Objective: Display error if a user attempts to execute or urgent execute a macro with invalid parameters.

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Fill in invalid parameters to the macro	LibertyMutual.db Input: Randomly tests (1 to n) fields are filled in incorrectly and do not follow specified regex and the rest (n to all) are filled in	A macro has been selected to update from the database
2.	User selects "Execute"		User is prompted "Invalid Parameters for Specified Macro"

		correctly given specified regex	

Postconditions: No macro has been selected to update from the driver table and the user must correct their input before attempting to update another macro. Invalid fields are emphasized in the UI.

Test Case ID: TC_Update_Macro_04 (References SRS, Section 2.2)

Objective: Display error if a user attempts to execute or urgent execute a macro with missing required parameters.

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to update from drop-down menu, field parameters are left empty field for the delete macro.	LibertyMutual.db Input: Randomly tests (1 to n) fields are not filled in and do not follow specified regex and the rest (n to all) are filled in correctly given specified regex	A macro has been selected to update from the database
2.	User selects "Execute"		User is prompted "Missing Required Parameters for Specified Macro"

Postconditions: No macro has been selected to update from the driver table and the user must correct their input before attempting to update another macro. Missing fields are emphasized in the UI.

Test Case ID: TC_Update_Macro_05 (References SRS, Section 2.2)

Objective: Display error if a user attempts to urgent execute a macro without required credentials.
Preconditions: User logged into system who is not an admin or developer with privileges, and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to update from drop-down menu, fill in empty fields for	LibertyMutual.db Input: Fields are filled in correctly and follow	A macro has been selected to update from the database

	parameters to the macro	the specified regex	
2.	User selects "Urgent Execute"		The credentials are not verified with the server and the user is given an error message "You do not have the required privileges, schedule a peer review instead with 'Execute'"

Postconditions: The user has received an error message and is back to specified update macro page with the option to "Execute" for peer review.

Delete Macro Test Cases

Test Case ID: TC_View_Delete_01 (References SRS, Section 2.3)

Objective: View specific update macro page

Preconditions: User is already logged in

Step Number	Step Description	Test Data	Expected Result
1.	Clicked delete in the body of the page		Show the dropdown menu of delete macro options
2.	Clicked specific delete macro option		Show the selected delete macro page

Postconditions: on selected delete macro page

Test Case ID: TC_Delete_Macro_01 (References SRS, Section 2.3)

Objective: Schedule a specific macro with required parameters to be removed by a peer

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to delete from drop-down menu.	LibertyMutual.db Input: Fields are filled	A macro has been selected to delete from the database

2.	User selects "Execute"	in correctly and follow the specified regex	User is prompted "Are you sure?"
3.	User selects "Yes"		The delete macro information is packaged into a delete object and logged into the peer review. User receives alert box saying request has been processed.

Postconditions: Macro is pending for approval in the peer review database. "Peer Review, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" is appended to the end of 'log.txt'.

Test Case ID: TC_Delete_Macro_02 (References SRS, Section 2.3)

Objective: Execute a macro by bypassing peer review

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to delete from drop-down menu.	LibertyMutual.db Input: Fields are filled in correctly and follow the specified regex	A macro has been selected to delete from the database
2.	User selects "Urgent Execute"		User is prompted "Are you sure?"
3.	User selects "Yes"		The user credentials are verified by the server. The delete macro information is packaged into a delete object and written to the driver tables. User receives alert box indicating the deletion of the macro.

Postconditions: The specified macro has been appended to the driver tables. "Override, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" is appended to the end of 'log.txt'.

Test Case ID: TC_Delete_Macro_03 (References SRS, Section 2.3)

Objective: Display error if a user attempts to execute or urgent execute a macro with invalid parameters.

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to delete from drop-down menu, fill in invalid parameters to the macro	LibertyMutual.db Input: Randomly tests (1 to n) fields are filled in incorrectly and do not follow specified regex and the rest (n to all) are filled in correctly given specified regex	A macro has been selected to delete from the database
2.	User selects "Execute"		User is prompted "Invalid Parameters for Specified Macro"

Postconditions: No macro has been selected to delete from the driver table and the user must correct their input before attempting to delete another macro. Invalid fields are emphasized in the UI.

Test Case ID: TC_Delete_Macro_04 (References SRS, Section 2.3)

Objective: Display error if a user attempts to execute or urgent execute a macro with missing required parameters.

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to delete from drop-down menu, field parameters are left empty field for the delete macro.	LibertyMutual.db Input: Randomly tests (1 to n) fields are not filled in and do not follow specified regex and the rest (n to all) are filled in correctly given specified regex	A macro has been selected to delete from the database
2.	User selects "Execute"		User is prompted "Missing Required Parameters for Specified Macro"

Postconditions: No macro has been selected to delete from the driver table and the user must correct their input before attempting to delete another macro. Missing fields are emphasized in the UI.

Test Case ID: TC_Delete_Macro_05 (References SRS, Section 2.3)

Objective: Display error if a user attempts to urgent execute a macro without required credentials.
Preconditions: User logged into system who is not an admin or developer with privileges, and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to delete from drop-down menu, fill in empty fields for parameters to the macro	LibertyMutual.db Input: Fields are filled in correctly and follow the specified regex	A macro has been selected to delete from the database
2.	User selects "Urgent Execute"		The credentials are not verified with the server and the user is given an error message "You do not have the required privileges, schedule a peer review instead with 'Execute'"

Postconditions: The user has received an error message and is back to specified remove macro page with the option to "Execute" for peer review.

Add Macro Test Cases

Test Case ID: TC_View_Add_01 (References SRS, Section 2.4)

Objective: View specific update macro page

Preconditions: User is already logged in

Step Number	Step Description	Test Data	Expected Result
1.	Clicked add in the body of the page		Show the dropdown menu of add macro options
2.	Clicked specific add macro option		Show the selected add macro page

Postconditions: on selected add macro page

Test Case ID: TC_Add_Macro_01 (References SRS, Section 2.4)

Objective: Schedule a specific add macro with required parameters to be removed by a peer

Preconditions: User is logged in and is on specific add macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select parameters for macro.	LibertyMutual.db Input: (1 to all) select from the drop down menu	A macro has been selected to add to the database with correct parameters
2.	User selects "Execute"		User is prompted "Are you sure?"
3.	User selects "Yes"		The add macro information is packaged into an add object and logged into the peer review. User receives alert box saying request has been processed.

Postconditions: Macro is pending for approval in the peer review database. "Peer Review, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" is appended to the end of 'log.txt'.

Test Case ID: TC_Add_Macro_02 (References SRS, Section 2.4)

Objective: Execute a macro by bypassing peer review

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select parameters for macro.	LibertyMutual.db Input: (1 to all) select from the drop down menu	A macro has been selected to add to the database with correct parameters
2.	User selects "Urgent Execute"		User is prompted "Are you sure?"
3.	User selects "Yes"		The user credentials are verified by the server. The add macro information is packaged into a add object and written to

			the driver tables. User receives alert box indicating the addition of the macro.

Postconditions: The specified macro has been appended to the driver tables. "Override, User Name, UserID, Macro Name, MacroID, Macro Parameters, Date" is appended to the end of 'log.txt'.

Test Case ID: TC_Add_Macro_03 (References SRS, Section 2.4)

Objective: Display error if a user attempts to execute or urgent execute a macro with invalid parameters.

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Fill in invalid parameters to the macro	LibertyMutual.db Input: Randomly tests (1 to n) don't select value from drop-down menu (n to all) select from the drop down menu	A macro has been selected to delete from the database, with invalid parameters.
2.	User selects "Execute"		User is prompted "Invalid Parameters for Specified Macro"

Postconditions: No macro has been selected to add to the driver table and the user must correct their input before attempting to add another macro. Invalid fields are emphasized in the UI.

Test Case ID: TC_Add_Macro_04 (References SRS, Section 2.4)

Objective: Display error if a user attempts to execute or urgent execute a macro with missing required parameters.

Preconditions: User is logged in and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Leave field required parameters empty field for the add macro.	LibertyMutual.db Input: Randomly tests (1 to n) fields are not filled in and do not follow specified regex and the rest	An add macro has been selected to delete from the database with empty parameters.
2.	User selects "Execute"		User is prompted "Missing Required

		(n to all) are filled in correctly given specified regex	Parameters for Specified Macro"

Postconditions: No macro has been selected to add from the driver table and the user must correct their input before attempting to add another macro. Missing fields are emphasized in the UI.

Test Case ID: TC_Add_Macro_05 (References SRS, Section 2.4)

Objective: Display error if a user attempts to urgent execute a macro without required credentials.
Preconditions: User logged into system who is not an admin or developer with privileges, and is on specific remove macro page.

Step Number	Step Description	Test Data	Expected Result
1.	Select a specific macro to add from drop-down menu, fill in empty fields for parameters to the macro	LibertyMutual.db Input: Randomly tests (1 to n) don't select value from drop-down menu (n to all) select from the drop down menu	A macro has been selected to delete from the database
2.	User selects "Urgent Execute"		The credentials are not verified with the server and the user is given an error message "You do not have the required privileges, schedule a peer review instead with 'Execute'"

Postconditions: The user has received an error message and is back to specified remove macro page with the option to "Execute" for peer review.