

Change request log for CR-3

1. Concept Location

Step #	Description	Rationale
1	First we ran the system	
2	We interacted with the system and checked if the CR given is right, using the details shown in screenshots and matching them with actual system.	To verify the change request.
3	Logged in to the system as admin	
4	Visited the Users page by clicking the small person icon on the top menu bar.	
5	Clicked on the admin in the list shown on the left on page /test/users.shtm	
6	Typed in new password and tried to change the password.	Could not change the password. This ended the verification of change request.
7	Installed the cscope and ctags for browsing the source code, To find the concept location in the source code.	We were familiar with these tools and had used them before for code browsing and navigation.
8	Created cscope files to use cscope on the src. <u>find . -name "*.java" > cscope.files</u>	
9	Created the tags file for all the java files in the main mango source directory.	This would help us in finding the definitions of various functions, trace calls for given functions statically in the code.
10	Started ccscope toolbox using <u>cscope -R</u> on src directory.	
11	Used the "Find this egrep pattern:" for searching various keywords in the source code.	To understand where the actual code needs to be changed.
12	Searched for string "New password" in the code.	Since the CR had something to do with updating the password.
13	Searched for string "password" in the code. Found multiple files.	
14	Found a function setPassword on egrep'ing password in src code.	
15	Started checking calls for setPassword using the Find functions calling this function: Functionality of cscope	To see where this function was being called from so that we find a place where password was being updated.
16	Found a file DatabaseAccess.java where an object of class User was being used to setPassword.	
17	Traced the object back to find definition of the User class.	
18	Found a file User.java and started checking if that is the file that needs to be changed.	
19	Found a function in User.java named setPassword.	Probably this could have been the place, since we were dealing with change of password.
20	Went to mango web page again and checked the error.	We weren't finding anything useful so though if we could do something else.
21	Searched for the query (update users set) shown in the text box using egrep	To check where the query string shown in error can be found in code. This was a major point in the process because this helped us reach where the actual query was used.

22	Found multiple queries starting with (update users set) in src/com/serotonin/mango/vo/User.java , so we tried finding all the instances in the file.	
23	Found a private static final String USER_UPDATE Which exactly matched the query shown in the log.	
24	Found that query contains all the fields that are seen on the webpage (/test/users.shtm) except that there is an extra field in the query named homeUrl	
25	Tried modifying the query and rerun the system.	To check if making changes to query changes the log shown in the system with updated query.
26	Found the exact concept location of where the change needs to be done.	Since changing the query changed the log in the system.

Time spent (in minutes): 65

Classes and methods inspected:

- src/com/serotonin/mango/web/mvc/form/LoginForm.java
- src/com/serotonin/mango/web/dwr/UsersDwr.java
- src/br/org/scadabr/vo/dataSource/opc/OPCDataSourceVO.java
 - public void validate(DwrResponseI18n response)
 - public void setPassword(String password)
- src/com/serotonin/mango/db/BasePooledAccess.java
- src/com/serotonin/mango/db/DatabaseAccess.java
 - public void initialize()
- src/com/serotonin/mango/vo/User.java
 - public void setPassword(String password)
 - public void recordLogin(int userId)
 - void updateUser(User user)

2. Impact Analysis

Step #	Description	Rationale
1	We used the functionality Find functions calling this function: to find which functions were calling updateUser(user)	To track if the user is being updated from multiple paces.
2	Found that only function saveUser is calling the function updateUser	
3	We then checked functions calling saveUser, to find that multiple functions were calling saveUser.	
4	We found that the initialize function did not set homeurl as opposed to in the update function.	This meant there is no change need to be done in the DatabaseAccess class as since no homeURL was being set there." Discarded – DatabaseAccess.java
5	We found that function DwrResponseI18n saveUser also did not set the homeUrl.	This meant there is no change need to be done in the UsersDwr class as since no homeURL was being set there." Discarded – UsersDwr.java
6	Similarly we discarded ImportTask.java since homeUrl was not being set there as well.	Discarded - ImportTask.java
7	This ended our impact analysis.	Since the homeurl was not being updated in the updateUser function or in functions calling it.

		Also, there is no place to update homeURL on the user interface or web page to update password. So removing homeUrl from the query should not impact any other classes.
--	--	---

Time spent (in minutes): 43

Classes and methods inspected:

- src/com/serotonin/mango/db/dao/UserDao.java
 - public void saveUser(final User user)
- src/com/serotonin/mango/db/DatabaseAccess.java
 - public void initialize()
- src/com/serotonin/mango/web/dwr/UsersDwr.java
 - public DwrResponseI18n saveUser(int id, String password, String email, String phone, int receiveAlarmEmails, boolean receiveOwnAuditEvents)
- src/com/serotonin/mango/web/dwr/beans/ImportTask.java
 - private void importUser(JsonObject userJson)
 - private void importUserPermissions(JsonObject userJson)

3. Prefactoring (optional)

No prefactoring was done in the code.

Time spent (in minutes): 0

4. Actualization

Step #	Description	Rationale
1	We created a new github branch from master.	To add changes on a new branch.
2	We updated the USER_UPDATE query string and removed the parameter homeUrl from the prepared statement.	This is the query which is giving error since the homeUrl is not being updated and is not being obtained from the User interface to update password.
3	We also updated the void updateUser(User user) function where the query was being executed and removed the parameter homeUrl which was being set there as well.	
4	We then compiled the code again using the provided steps 1. ant fulldeploy and 2. ant reload	To check if the changes work.
5	Logged in to the system using admin and navigated to the users page by clicking on the person icon on top left menu bar.	
6	Clicked on admin and updated the password.	To check if password can be changed for admin or the error still persists.
7	There was no error and there was a message User saved.	
8	We logged out using the logout option on menu bar.	To try checking if the password really changed.
9	Tried logging in with old password and it did not work.	
10	Tried logging in with the changed password and we were able to login.	

11	We committed the changes to github.	Just in case if we need to revert the change.
----	-------------------------------------	---

Time spent (in minutes): 15

Inspected Classes:

- src/com/serotonin/mango/db/dao/UserDao.java

Classes changed:

- src/com/serotonin/mango/db/dao/UserDao.java
 - private static final String USER_UPDATE
 - void updateUser(User user)

5. Postfactoring (optional)

No postfactoring was done in the code.

We found out that the homeUrl was not being set anywhere. This would require some more efforts to check and remove the dead code/ functionality if home url is not needed or also it could be set at proper places and keep an option for user to update. It can be done in a new change request.

Time spent (in minutes): 10

- src/com/serotonin/mango/vo/User.java
 - public void setHomeUrl(String homeUrl)
- src/com/serotonin/mango/web/dwr/MiscDwr.java
 - public void setHomeUrl(String url)

6. Validation

Step #	Description	Rationale
1	Before change of code: Functional Test: We tried changing the password of admin from the update user page.	We were not able to login and got the expected error message. Test Passed.
2	After change of code: Functional Test: We tried changing the password of admin from the update user page. Input –New password: test	We were able to change the password successfully and able to login using the new password. Test Passed.
3	Functional Test: Input –New password: test12345 Expected Output–New password: test12345	Test Passed.
4	Functional Test: Input – Leaving the password field blank. Expected Output – Should not affect the password field. The old password should remain same as user does not intend to change the password.	The old password should remain same as user does not intend to change the password. Test Passed.
5	Functional Test: Input – New Phone – 123123123 Expected Output – New Phone – 123123123	The new phone should get changed. Checked by logging out and going to user page again and the new phone appeared. Test Passed.
6	Functional Test:	Test Passed.

	Tried changing Send alarm emails field to all the possible values.	
7	Functional Test: Tried changing all update values possible in the User details page.	Changing the query should not affect the old functionality. User should be able to perform all update options present. All functionalities worked as before changing the code. Test Passed.

Time spent (in minutes): 25

7. Summary of the change request

Phase	Time (minutes)	No. of classes inspected	No. of classes changed	No. of methods inspected	No. of methods changes
Concept location	65	6	0	6	0
Impact Analysis	43	4	0	5	0
Prefactoring	0	0	0	0	0
Actualization	15	1	1	1	1
Postfactoring	10	2	0	2	0
Verification	25	1	0	1	0
Total	148	9	1	7	1

8. Conclusions

The actual change being small and easy, it took us a bit time to find the concept location. We used cscope and ctags for finding for navigating through the source code as we were already familiar with these tools. Also used vim as the text editor to change the code. We performed functional tests by trying to change the password and checked if other functionalities worked as before on the users update page. This required testing before and after the change as well to compare.

We did not perform refactoring of any type but found that the home url is not being set anywhere and used correctly. However, we did not change it since that would require some more efforts, time and that can be done by creating a new change request.