Software Engineering Group Project
Test Specification

Author: L. Jones, T. Oram, W. Jones, T. MIlls
Config. Ref.: SE-12-TS
Date: 2015-11-12
Version: 1.0
Status: Release

Department of Computer Science,
Aberystwyth University,
Aberystwyth,
Ceredigion, SY23 3DB,
U.K.

CONTENTS

1 INTRODUCTION

1.1 Purpose of this document

The purpose of this document is to outline how we aim to enforce good testing practice, and as such contains a 'Test Specification' which demonstrates how to test against the different requirements as specified in the requirements specification[1], as well as displaying the criteria that must be met in order to pass tests.

1.2 Scope

This document specifically covers the 'Test Specification' section of the Test Procedure Standards Quality Assurance Document[2]

This document should be read by all project members. It is assumed that the reader is already familiar with the Test Procedure Standards [2].

1.3 Objectives

The objectives of this document are as follows:

- Identify where a test will be necessary against the requirements specificationse.qa.rs and give it a unique ID.
- Explain the exact function/requirement that is being tested.
- Explain how a user would undertake a test, as well as the desired output if successful.
- Explain the criteria that must be met by the test in order for it to be deemed as "Passed".

2 TEST SPECIFICATION

Test Reference	Reference Requirement	Test Content	Input	Output	Pass Criteria
SRV-001	FR1	To check that the system can create two pieces of information; full name and email address	In an SQL command line insert into the team member table	The table should now contain the entered data	The entered data should be returned when a query is run.
SRV-002	FR1	To check that the database can support the updating of team member data	In an SQL command line; update the data in the team member table	The table should now contain the updated data	The data will be updated to the new data that was entered
SRV-003	FR1	To check that the system supports deletion of team members	In an SQL command line; delete the data in the team member table	The table should not include the member that was deleted	The table should not have a record of that team member
SRV-004	FR2	The system must be able to support the creation of a task	In an SQL Command Line; Create a task setting the title, whom it is allocated to, start and end dates, the task elements and the task status and then go to the "display tasks" page	The task table should now have one more record	The task will have been created and will consist of all the data that was entered in before
SRV-005	FR2	The system must be able to support the updating of a task	In an SQL command line; update the task data to have a task status of 'allocated' and an end date of 21/10/2016	The data should be updated	The task should have the new end date and status
SRV-006	FR2	The system must be able to support the deletion of tasks	In an SQL command line; Delete a record in the task table	The task table should not include the record that was removed from the database	The task will be deleted from the database

MAN-001	FR3	TaskerMAN should support the creating of team member data	In the Create user page of taskerMAN create a new user with name "John Smith" and email "js@aber.ac.uk"	The Show users page in TaskerMAN will now include the user "John Smith"	All data entered should be correct and the show users page displays the member
MAN-002	FR3	TaskerMAN should support the updating of a team member	UUnder the edit user page of TaskerMAN edit users name to "Tom Smith"	The show users page in TaskerMAN should now have the updated task member	The task member will now have a name of "Tom Smith"
MAN-003	FR3	TaskerMAN should support the deletion of a team member	Under the edit user page of TaskerMAN delete the user with the name "Tom Smith"	The user will be deleted from the database	The show users page in TaskerMAN should no longer include the team member that was deleted
MAN-004	FR4	TaskerMAN should support the creation of task data	Under the create task page of TaskerMAN create a task where the task title is "Make coffee" with a start date of ?20/10/2015? and an end date of '21/10/2015' also set the member assigned and also the tasks status	The task should be added to the database	The show tasks page in taskerMAN will show the task entered and should have the exact data entered
MAN-005	FR4	TaskerMAN should support the updating of task data	Under the edit task page in TaskerMAN select any task to update and update the task status to 'completed'	The database should be updated	The show tasks page in TaskerMAN will show the updated task and its task status should now be 'completed'
MAN-006	FR4	TaskerMAN should support the deleting of task data	Under the edit task page in TaskerMAN, select any task and then delete it	The selected task will be deleted from the database	The show tasks page in TaskerMAN should no longer display the task that was deleted

MAN-007	FR5	TaskerMAN should support the re-allocating of a task	Under the edit task page in TaskerMAN select any task to edit. Next; change the member allocated to "John Smith"	The task should be updated in the database	The show tasks page in TaskerMAN should now show the task with "John Smith" allocated to it
MAN-008	FR6	TaskerMAN should support the abandoning of a task	Under the edit task page in TaskerMAN select any task to edit. Next; change the task status to "Abandoned"	The task should be updated in the database	The show tasks page in TaskerMAN should now show the task with a task status of 'Abandoned'
MAN-009	FR7	TaskerMAN should support the viewing of tasks	On TaskerMAN click "View Tasks"	A query should be sent to list all tasks in the database	A list of every task will be displayed
MAN-010	FR7	TaskerMAN should support the filtering of tasks dependant on allocated team member	On TaskerMAN in the "view tasks" page select the option to filter by team member and then type "John Smith"	A query should be sent to list all the tasks that are allocated to "John Smith"	The list of tasks that are displayed should only be allocated to "John Smith"
MAN-011	FR7	TaskerMAN should support the filtering of tasks dependant on the task status	On TaskerMAN in the "view tasks" page select the option to filter by task status and select to show all the tasks that are 'completed'	A query should be sent to list all the tasks that have a task status of 'completed'	The list of tasks that are displayed should only have a task status of 'completed'
CLI-001	FR8	TaskerCLI should be able to identify the member upon logging in	Upon login, type an admins username and password and then press the login button Username: jsmith Password: password;	A query should be sent to the database to locate the user with the data entered	The user will then be logged in and on the main page of TaskerCLI the current user will say "logged in as "jsmith""

CLI-002	FR8	TaskerCLI should store the login details locally for future use	Upon login select the "remember me" checkbox and enter the details: UserName: Jsmith Password: password	Will run a query that will return a record with matching credentials and will be stored locally	When the program is restarted the user will be automatically logged in
CLI-003	FR8a	TaskerCLI should be able to store data on the users computer locally for the user that is logged in	Login to a user (this will synchronise)	After synchronisation the program should save all the tasks that are allocated to them locally to a text file (located in the users home directory)	The text file will now display a table showing the tasks that were allocated to the member logged in
CLI-004	FR9	TaskerCLI should be able to synchronise to get all the tasks for the user that is logged in at the time and should only be ones that have a task status of 'allocated'	Login to a user that has tasks allocated to them with a status of 'allocated'. Then press the synchronisation button	A query will be ran to return all of the tasks that are allocated to the user and those tasks will be stored in a text file locally in the users home directory	The text file will contain a list of tasks that will be identical to those stored in the database that are allocated to the user that is logged in at the time and has a task status of 'allocated'
CLI-005	FR9	Synchronisation should support the deletion of completed/abandoned tasks	Login to a user that already has a text file containing tasks allocated to them with a task status of 'allocated' and then edit one of the tasks so that its task status is now 'completed'. Also edit one of the tasks so that its status is 'abandoned'	A query will be ran to return all of the tasks that are allocated to the user and those tasks will overwrite the existing text file (removing any completed/abandoned tasks)	The text file will not display the task that was set to ?completed? and it will also not display the task that was set to 'abandoned'

CLI-006	FR10	TaskerCLI must be able to support local editing of tasks. (Task status)	Select a task from the list of tasks and then select to edit that task. Change the tasks status to 'completed' then proceed to view tasks	The task will have been updated in the database	In the show tasks panel the task that was edited will now have a task status of 'completed'
CLI-007	FR10	TaskerCLI must be able to support the local editing of tasks. (task element note)	Select a task from the list of tasks and then select to edit that task. Edit one of the task elements notes to something different	The task element should have been updated in the database	The show tasks panel will now display the task with the updated task note
CLI-008	FR11	Synchronisation must happen on start up	After logging in TaskerCLI will start and synchronisation will happen at this time. Check the local files to see if it has displayed all the current allocated tasks for that member in a text file	The TaskerCLI should synchronise with the database and the local files will be updated	The local files should be updated straight after login
CLI-009	FR11	Synchronisation must happen before editing a task	Edit a task in the database with SQL by setting its task status to 'completed'. Next; select the task (this should not show the updated task status yet) and click edit task	TaskerCLI should synchronise with the database and display the task with the updated task status that were input using SQL	Within the field boxes on the form in the edit task panel of TaskerCLI the newly updated task will be displayed showing the task status as 'completed'

CLI-010	FR11	Synchronisation must happen after editing a task	Login to a user that has tasks allocated to them and select to edit one of the tasks and change its task status to 'completed' Press update. Proceed to check the database for this task to see if it has been updated by running a query	TaskerCLI should synchronise with the database and the new task will be shown in the database after the SQL query is ran	TaskerCLI will have synchronised and the query will return the field with the updated task status
CLI-011	FR11	TaskerCLI should synchronise with the database every 5 minutes of it being logged on	Login to TaskerCLI and record the time of start. Update a task in the database whilst waiting for the 5 minutes to pass. Next: Refresh the show tasks panel after 5 minutes have passed from the start time	After 5 minutes TaskerCLI should synchronise with the database and store the new updated task in the show tasks panel	The updated task will be displayed in the show tasks panel after refreshing when 5 minutes have passed
UI-001	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to create a user (for themselves) in TaskerMAN	User should be able to create a team member without asking for help	It was obvious to the user
UI-002	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to edit a team member In TaskerMAN	User should be able to edit a team member without asking for help	It was obvious to the user
UI-003	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to delete a team (not themselves) member in TaskerMAN	User should be able to delete a team member without asking for help	It was obvious to the user

UI-004	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to create a new task in TaskerMAN and set the task to be allocated to them	User should be able to create a new task without asking for help	It was obvious to the user
UI-005	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to edit the task that is allocated to them to have a task status of 'completed' in TaskerMAN	User should be able to edit a new task without asking for help	It was obvious to the user
UI-006	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to delete a task (not the task that was allocated to them) TaskerMAN	User should be able to delete a new task without asking for help	It was obvious to the user
UI-007	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to view members of the group	User should be able to view the team members in the group without asking for help	It was obvious to the user
UI-008	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	non-developer of the group try to view the tasks	User should be able to view the tasks in the group, without asking for help	It was obvious to the user
UI-009	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to Filter tasks that are allocated to them	User should be able to see the task that was allocated to them, without asking for help	It was obvious to the user
UI-010	IR1	The user interfaces of TaskerMAN should be easy to use by regular computer users	A non-developer of the group try to Filter tasks that are completed	User should be able to see the task that they created (that has a task status of 'completed')	It was obvious to the user

UI-011	IR1	The user interfaces of TaskerCLI should be easy to use by regular computer users	A non-developer of the group try to login to TaskerCLI	User should be able to login without asking for help	It was obvious to the user
UI-012	IR1	The user interfaces of TaskerCLI should be easy to use by regular computer users	A non-developer of the group try to view tasks in TaskerCLI	User should be able to navigate to be able to view tasks without asking for help	It was obvious to the user
UI-013	IR1	The user interfaces of TaskerCLI should be easy to use by regular computer users	A non-developer of the group try to edit a task in TaskerCLI	User should be able to edit a task without asking for help	It was obvious to the user
UI-014	IR1	The user interfaces of TaskerCLI should be easy to use by regular computer users	A non-developer of the group try to use the search bar to search for a task	User should be able to search for a task using the search bar without asking for help	It was obvious to the user
UI-015	IR1	The user interfaces of TaskerCLI should be easy to use by regular computer users	A non-developer of the group try to filter tasks by those allocated to them	User should be able to filter tasks that were allocated to them without asking for help	It was obvious to the user
UI-016	IR1	The user interfaces of TaskerCLI should be easy to use by regular computer users	A non-developer of the group try to synchronise	User should be able to synchronise TaskerCLI with the database without asking for help	It was obvious to the user
PER-001	PR1	Program should respond to user input in a minimum of a second	Enter a user's credentials into the login page of TaskerCLI and click enter	The page should log in in less than a second	The page logged in in less than a second
PER-002	PR1	Program should respond to the user input in a minimum of a second	Press show tasks in TaskerCLI	The tasks should be displayed in less than a second	The tasks were displayed in less than a second
PER-003	PR1	Program should respond to the user input in a minimum of a second	Enter data for a new team member in TaskerMAN	The member should be visible within one second	The user was visible within one second

PER-004	PR1	Program should respond to the user input in a minimum of a second	Enter data for a new Task in TaskerMAN	The task should be visible within one second	The task was visible within one second
PER-005	PR2	TaskerCLI should run on any machine supporting Java	Run TaskerCLI on a Linux machine	Program should execute	Program executed
PER-006	PR2	TaskerCLI should run on any machine supporting Java	Run TaskerCLI on a Windows machine	Program should execute	Program executed
PER-007	PR2	TaskerCLI should run on any machine supporting Java	Run TaskerCLI on a Mac	Program should execute	Program executed
PER-008	PR2	TaskerSRV must run on a suitable web server	Test connection to database using JDBC by logging into TaskerCLI	Should connect successfully	JDBC connected to the database successfully
PER-009	PR2	TaskerMAN should be deployable on an apache web server	Deploy TaskerMAN to an apache web server	Should work correctly	Worked correctly
DES-001	DC1	TaskerCLI must be written in Java	Check over code make sure it is written in the correct standard of Java	Should be written in the correct standard	Code was valid
DES-002	DC2	TaskerMAN must be written in any Server side language	Check over code to make sure it is written in the correct standard of PHP and JS	Should be written in the correct standard	Code was valid
DES-003	DC2	The system should be able to work with a constraint of 20 members with 30 allocated tasks	Enter into the database 20 members and 30 allocated tasks. Use basic functionality of the program and check for lag	There should be little lag (still working within one second from user input)	No/little latency

REFERENCES

[1] *Software Engineering Group Projects* Requirements Specifications. N. W. Hardy, SE.QA.RS. 1.2 Release.

[2] *Software Engineering Group Projects* Test Procedure Standards C. J. Price, N.W.Hardy and B.P.Tiddeman, SE.QA.06. 1.8 Release.

DOCUMENT HISTORY

Version	CCF No.	Date	Changes made to Document	Changed by
1.0	N/A	2015-11-12	Initial creation for LaTeX	L. Jones