Software Engineering Group Project
Test Specification

Author:	L. Jones, T. Oram, W. Jones, T. MIlls
Config. Ref.:	SE-12-TS
Date:	2016-02-14
Version:	1.1
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
MAN-001	FR3	TaskerMAN should support the creation 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 team member's page displays js@aber.ac.uk
MAN-002	FR3	TaskerMAN should support the creation of team member data	In the create team member page of TaskerMAN try to enter in "hello" as the email address.	Client side form validation should execute.	Form validation should notify user that it is not a valid email address.
MAN-003	FR3	TaskerMAN should support the creation of team member data	In the Create team member page of TaskerMAN, create a team member with their first name as "DROP TABLE User"	An SQL query will be executed with the first name set as "DROP TABLE User?.	Remote database will still be intact if protection against SQL injection is correctly set up.
MAN-004	FR3	TaskerMAN should support the updating of team member data	Under the edit team member page of TaskerMAN select to edit "js@smith.com" and change its name to "Tom Smith".	The show team members page in TaskerMAN should now have the updated team member	Form validation should notify user that it is not a valid email address.
MAN-005	FR3	TaskerMAN should support the updating of team member data.	Under the edit team member page of TaskerMAN edit "js@smith.com"'s email to be "hello".	Client side form validation should execute.	The show tasks page in TaskerMAN will show the updated task with it's new title
MAN-006	FR3	TaskerMAN should support the deletion of team member data.	Under the edit team member page of TaskerMAN delete the team member with the name "Tom Smith"	The team member will be deleted from the database.	The show team member's page in TaskerMAN should no longer include the team member "js@smith.com"

MAN-007	FR3	TaskerMAN should support the creation of task data.	In the create team Member page of TaskerMAN create a new team member with name “John Smith” and email “js@aber.ac.uk” (Test subject needed)	Query will be executed to Insert into the team member table the new team member.	Database should now include the team member “js@aber.ac.uk.”
MAN-008	FR4	TaskerMAN should support the updating of task data	Under the create task page of TaskerMAN create a task with: Title = “make coffee”, Start Date =“30/01/2016”, End Date = “1/02/2016”, TaskElement1 title = “get Cup”, TaskElement2 title= “grind beans”, and allocate it to “js@smith.com”	The task should be added to the remote database.	The show tasks page in TaskerMAN will show the task entered and should have the exact input data and have a Task Status of ?ALLOCATED? and will be allocated to “js@smith.com?.
MAN-009	FR4	TaskerMAN should support the creation of task data	Under the create task page of TaskerMAN create a task with: Title = “make coffee”, Start Date =“30/01/2016”, End Date = “1/01/2000”, TaskElement1 title = “get Cup”, TaskElement2 title= “grind beans”, and allocate it to “js@smith.com”	Client Side form validation should execute.	User should be notified that they entered an end date that was before the start date.

MAN-010	FR4	TaskerMAN should support the creation of task data (Testing against SQL Injection)	Under the create task page of TaskerMAN create a task with: Title = "DROP TABLE Tasks", Start Date ="30/01/2016", End Date = "1/02/2016", TaskElement1 title = "get Cup", TaskElement2 title= "grind beans", and allocate it to "js@smith.com"	SQL Query that partly contains the String "DROP TABLE Tasks" will be executed	Database should still be intact.
MAN-011	FR4	TaskerMAN should support the updating of task data.	Under the edit task page in TaskerMAN select the "Make Coffee" task. Choose to add an extra task element: Title= "pour water".	The task should be updated in the database	The show tasks page in TaskerMAN will show the updated task with an extra Task Element.
MAN-012	FR4	TaskerMAN should support the deletion of task data.	Under the edit task page in TaskerMAN, select the task "make coffee". Delete this task.	The selected task will be deleted from the database.	The show tasks page in TaskerMAN should now not include the task "make coffee" as it should've been removed from the remote database.

MAN-013	FR5	TaskerMAN should support the re-allocation of a task	Create a new team member, first name: “testy”, last name “mctesty”, email: “test@test.test” and a password of “123”. Create a task with: Title = “make coffee”, Start Date =“30/01/2016”, End Date = “1/02/2016”, TaskElement1 title = “get Cup”, TaskElement2 title= “grind beans”. Allocate it to test@test.test. Go to edit tasks page, select the task and choose to re-allocate it to “js@smith.com”.	The task should be updated in the remote database.	The show tasks page in TaskerMAN should now show the task with “js@smith.com” allocated to it.
MAN-014	FR6	TaskerMAN should support the abandonment of a task	Under the edit task page in TaskerMAN select to edit the task “Make coffee”, change its task status to “Abandoned”	The task should be updated in the remote database.	The show tasks page in TaskerMAN should now show the task with a task status of “Abandoned”.
MAN-015	FR7	TaskerMAN should support the viewing of tasks.	Create a new task with: Title = “Hey Im a task look at me”, Start Date = “12/02/2016”, End Date = “12/02/2016”, TaskElement1 = “look at me pls” and allocate it to “js@smith.com” (This is so that at least one task exists on the remote database). On TaskerMAN click on the “View Tasks” page	A query should be sent to list all tasks in the database.	The task ”Hey Im a task look at me” will be displayed, along with any other tasks should they exist in the remote database

MAN-016	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	A query should be sent to the remote database to list all the tasks by team member	The list of tasks that are displayed should be displayed by member in alphabetical ascending order.
MAN-017	FR7	TaskerMAN should support the filtering of tasks dependant on task status.	On TaskerMAN in the “view tasks” page select the option to filter by task status	A query should be sent to the remote databse to list all the tasks by task status	The list of tasks that are displayed should be displayed by task status in alphabetical ascending order.
CLI-001	FR8	TaskerCLI should be able to identify the member upon logging in	Upon login, enter the user credentials: Username: “jsmith@smith.com”, Password: “123”.	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 “Tom Smith””
CLI-002	FR8	TaskerCLI should be able to identify the member upon logging in	Upon login enter in the credentials: Username = “bob”, Password = “1”	Authenticator will check to see if the user in the database.	User should be notified that their credentials were incorrect.
CLI-003	FR8	TaskerCLI should be able to identify the member upon logging in	Upon login enter in the credentials: Email= “DROP TABLE User”	A query will be run that partly contains the String “DROP TABLE User”.	The remote database should still be intact.
CLI-004	FR8	TaskerCLI should store the login details locally for future use.	Upon login enter in the credentials: Userame = “js@smith.com” and the password = “123”, then logout.	The users details will be stored in the local SQLite database (password should be hashed).	“js@smith.com” should have their credentials stored in the local database (password also hashed).
CLI-005	FR8a	TaskerCLI should be able to store data on the user’s computer locally for the user that is logged in.	Log in to the user “js@smith.com” (password: 12345).	This should synchronise and store all data (locally) associated with “js@smith.com”.	The Local SQLite Database should contain the tasks for “js@smith.com”

CLI-006	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”	Create a new task on TaskerMAN with: Title = “test code”, Start date = “30/01/2016”, End date = “01/02/2016”, TaskElement1 = “module testing”, TaskElement2 = “system testing”. Allocate it to j“s@smith.com”. On TaskerCLI log into the user “js@smith.com”	This should synchronise with the remote database and display the list of tasks including the newly assigned task to “js@smith.com”.	JTable displays added task.
CLI-007	FR10	Synchronisation should support the deletion of completed/abandoned tasks.	Using TaskerMAN, select the task “make coffee” and set its Task Status = “ABANDONED”. Login to “js@smith.com” (on TaskerCLI)	Upon logging in, TaskerCLI should synchronise with the remote database and no longer contain the “make coffee” task.	The table will no longer contain the “make coffee” task.
CLI-008	FR9	Synchronisation should support the deletion of completed/abandoned tasks.	Log into “js@smith.com” on TaskerCLI, select the task “test code” and select complete task.	After editing the task, TaskerCLI will synchronise with the remote database and should no longer display the “test code” task.	The completed task should be removed from the table.

CLI-009	FR10	TaskerCLI must be able to support the local editing of tasks (Task element comments)	On TaskerMAN create a task with: Title = “Do chores”, Start Date = “30/01/2016”, End Date= “1/02/2016”, TaskElement title = “Make a list of chores”, TaskElement2 title= “Find way of procrastinating”. Allocate it to “js@smith.com”. Then login to TaskerCLI using “js@smith.com”. Edit the task “Do chores” and enter the comment “go for a pint” on the Task Element “Find way of procrastinating”.	TaskerCLI should synchronise with the database and update comment for the Task Element.	Upon selecting the task from now on the comment “go for a pint” should remain associated with the element “Find way of procrastinating”.
CLI-010	FR10	TaskerCLI must be able to support the local editing of tasks (Task element comments)	Login to TaskerCLI with the user “js@smith.com”, edit the task “Do chores” and enter the comment “DROP TABLE Tasks” on the Task Element “Make a list of chores”.	An SQL query will be executed which will partly contain the String “DROP TABLE Tasks”.	Remote database should still be intact.
CLI-011	FR11	Synchronisation must happen on startup	On TaskerMAN proceed to edit tasks, edit the task “Make a list of chores” by changing the Task Element “Find way of procrastinating” to “do chores tomorrow instead”. Login to TaskerCLI with “js@smith.com”	TaskerCLI should synchronise with the database after logging in and update the Task Element.	After logging into TaskerCLI select the task “Make a list of chores”. The Task Element “Find way of procrastinating” should now be “do chores tomorrow instead”.

CLI-012	FR11	Synchronisation must happen before editing a task.	Login to“ js@smith.com” on TaskerCLI. On TaskerMAN update the task “Do chores” so that the Task Element “Make a list of chores” is now “Think about chores”. Then select the task on TaskerCLI.	After selecting the task, TaskerCLI should synchronise with the remote database and update the Task Element.	After logging into TaskerCLI select the task “Make a list of chores”. The Task Element should now be called “Think about chores”.
CLI-013	FR11	Synchronisation must happen after editing a task.	Login to“ js@smith.com” on TaskerCLI. Select the task “Do chores” and add the comment “think harder” to the task element “Think about chores”, then close the task.	After selecting the task, TaskerCLI should synchronise with the remote database and update the Task Element.	The updated task element and comment should now be visible on TaskerMAN.
CLI-014	FR11	TaskerCLI should synchronise with the database every 5 minutes of it being logged on.	Login to“ js@smith.com” on TaskerCLI. On TaskerMAN edit the task “Think about chores” and set it’s status to “abandoned”. Go back to TaskerCLI and wait for automatic sunchronisation to occur.	After 5 minutes TaskerCLI should synchronise.	The task should no longer be visible on TaskerCLI after synchronisation.
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	Team member has been created without any difficulty
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	The team member data can be edited by the user without any difficulty

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	The team member data should be deleted without any difficulty
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	A task should be created by the user and allocated to themselves without any difficulty
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	The user should be able to change the status of the task to "Completed" without any difficulty
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	The user should be able to delete the task without any difficulty
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	The user should be able to view the team members in the group without any difficulty
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	The user should be able to view the tasks allocated to group members without any difficulty
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	The user should be able to filter tasks that are allocated to them without any difficulty
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')	The user should be able to filter tasks that have been completed without any difficulty

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	The user should be able to login to TaskerCLI without any issue
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	The user should be able to view the tasks and their respective task elements without any difficulty
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	The user should be able to edit a task without any difficulty
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	The user should be able to use the search bar to search for a specific task without any difficulty
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	The user should be able to filter the tasks without any difficulty
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

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
1.1	N/A	2016-14-02	Updated by Tom M based on Nigel's feedback, final version	L. Jones