

METEOCAL PROJECT

Test Case Manual

Authors: Claudio Sanna Walter Samà

Contents

1	Intr	roduction	3
2	Aut	tomatic Test	4
	2.1	EventTest	4
	2.2	NotificationTest	5
	2.3	SchedulerControllerTest	5
	2.4	SearchTest	6
	2.5	SettingsTest	7
	2.6	ShowEventControllerTest	8
	2.7	TimeControllerTest	9
	2.8	UserRegistrationTest	10
3	Ma	nual Testing	1

Chapter 1

Introduction

This document will provide information about all the test that were made to test our Meteocal web application. We used two different approach:

- An automatic one, using the testing tool provided by java (Junit, Mockito).
- Manual test, that were done by our self, putting different input and reporting the received output.

Chapter 2

Automatic Test

2.1 EventTest

Test Case Name	OkWithBadInviteTest
Goal	Test event, badConditions and participant creation
Input	An Event with correct input, a badCondition
Expected output	Creation of the event, the badCondition and the partic-
	ipant.
Output obtained	The output expected.
Additional Comments	none

Test Case Name	OkWithBadInviteTest
Goal	Test The method "dateAndTimeInTheMiddleCreate".
Input	Any type of possible input to the method "dateAnd-
	TimeInTheMiddleCreate".
Expected output	A false return value.
Output obtained	The output expected.
Additional Comments	The method "dateAndTimeInTheMiddleCreate" checks
	the data consistency of not overlapping events; since no
	event was previously created the return is always false
	for this test.

Test Case Name	NotDateError
Goal	Test the data input error.
Input	An Event with starting date in the past
Expected output	A NullPointerException.
Output obtained	The output expected.
Additional Comments	The exception is not thrown during the executing of the
	application but requestContext.execute is called; since
	in the test this will cause a NullPointerException we
	forced this to happen.

2.2 NotificationTest

Test Case Name	is Correctly V is ualized Test
Goal	Test The methods "isVisualized" and "isNotVisualized"
	used to check whether if a notification has been visual-
	ized or not.
Input	A notification that has not been visualized.
Expected output	The String "none" and "display" respectively fort "isVi-
	sualized" and "isNotVisiaulized".
Output obtained	The output expected.
Additional Comments	The string are used in the xhtml context to show and
	hide the element dynamically.

2.3 SchedulerControllerTest

Test Case Name	dataMergeTest
Goal	Test "dataMerge".
Input	Two different Date.
Expected output	A new Date with the date (year, month, day) of the first
	Date and the time of the second Date.
Output obtained	The output expected.
Additional Comments	none.
•	

Test Case Name	loadPublicCalendarTest
Goal	Check if is Public (that indicates if the calendar created
	is public or of the logged user) is set to true during the
	loading of a public calendar.
Input	A schedulerController.
Expected output	A true value of isPublic.
Output obtained	The output expected.
Additional Comments	none.

Test Case Name	loadOwnCalendarTest
Goal	Check if is Public (that indicates if the calendar created
	is public or of the logged user) is set to false during the
	loading of the logged user's calendar.
Input	A schedulerController.
Expected output	A false value of isPublic.
Output obtained	The output expected.
Additional Comments	none.

2.4 SearchTest

Test Case Name	search
Goal	Check "PrivateCalendar" (that indicates if the searched
	calendar created is public or private).
Input	A private calendar, a shared calendar, a public calendar.
Expected output	"PrivateCalendar" will assume true value for a shared
	and public calendar, otherwise false.
Output obtained	The output expected.
Additional Comments	none.

2.5 SettingsTest

Test Case Name	initTest
Goal	Check if all the current user setting are loaded.
Input	A user.
Expected output	The settings page and bean will take the information of
	the user stored in the database correctly.
Output obtained	The output expected.
Additional Comments	none.

Test Case Name	checkPasswordTest
Goal	Check the correct insertion of the old password.
Input	A null password and a correct password.
Expected output	A true value to "checkPassword" when the correct pass-
	word is inserted, otherwise false
Output obtained	The output expected.
Additional Comments	none.

Test Case Name	changeMailPassTest
Goal	Check if the email is changed.
Input	The logged user.
Expected output	The user will have the same settings as before except for
	the new email.
Output obtained	The output expected.
Additional Comments	none.

Test Case Name	changePrivacyToSharedt
Goal	Check if the calendar's privacy is changed.
Input	The logged user.
Expected output	The user will have the same settings as before except for
	the calendar privacy.
Output obtained	The output expected.
Additional Comments	none.

2.6 ShowEventControllerTest

Test Case Name	getDateTest									
Goal	Check if the bean gets the correct information from the									
	database.									
Input	An event with some participants.									
Expected output	The bean will have the correct date and time of the									
	event.									
Output obtained	The output expected.									
Additional Comments	none.									

Test Case Name	weatherImagesTest
Goal	Check if the bean gets the correct information from the
	database and show the correct image about the weather.
Input	An event with some participants and the weather asso-
	ciated to it.
Expected output	The bean will have the correct information about
	weather and will show the correct image.
Output obtained	The output expected.
Additional Comments	none.

Test Case Name	responseNotification						
Goal	Check if a new notification is created when a new re-						
	sponse is set.						
Input	An event with some participants and a new repsonse to						
	an invitation.						
Expected output	The bean will create a new notification.						
Output obtained	The output expected.						
Additional Comments	none.						

Test Case Name	isInvitedTest
Goal	Check if the bean gets the correct information from the
	database.
Input	An event with some participants.
Expected output	The bean will have the correct information about the
	participants (name and response).
Output obtained	The output expected.
Additional Comments	none.

2.7 TimeControllerTest

Goal Check the method "checkPrecipitations". Input Differente weather with different badconditions layer and precipitations value. Expected output The method will return true if (due to project specifi-
and precipitations value.
Expected output The method will return true if (due to project specifi-
cation) the badConditions constraint are violated oth-
erwise false.
Output obtained The output expected.
Additional Comments none.

Test Case Name	checkCondition
Goal	Check the method "checkPrecipitations".
Input	Different weather with different badconditions
Expected output	The method will return true if (due to project specifi-
	cation) the badConditions constraint are violated oth-
	erwise false.
Output obtained	The output expected.
Additional Comments	none.

${\bf 2.8}\quad {\bf User Registration Test}$

Test Case Name	${\it newUsersOkAndRepeated}$
Goal	Check if a user is correctly registerd into the system.
Input	A new user.
Expected output	The user will be added to the database and a new cal-
	endar will be created.
Output obtained	The output expected.
Additional Comments	none.

Test Case Name	new Users Ok And Repeated							
Goal	Check if a user is not created when an username already							
	used is inserted.							
Input	An user.							
Expected output	The user will be not added to the database and an error							
	will be shown.							
Output obtained	The output expected.							
Additional Comments	none.							

Test Case Name	newUserNull
Goal	Check if an exception is thrown when a null user name
	is inserted.
Input	An user with no user name.
Expected output	An exception will be thrown.
Output obtained	The output expected.
Additional Comments	none.

Chapter 3

Manual Testing

For manual testing we test our application by inserting various data (both correct and wrong) in order to test the correct functionality of our application to different input; all the result obtained are reported in the table in the next page.

Because the only method that needed a deep testing phase is the add event function, we created a set of test case only for that event. We reported different situation (behaviour of the application to null value inserted, to incorrect value, and to correct value) and the output provided.

, An	Any valid value																					
	no value																					
title I	location	city	starting date	ending date	starting time	ending time	repeat	untilidate	description	baconditions	layer	temp	prec	slidetemp	alldeprec	color	privacy	Invitation	Invitated user	outputmessage in dialog	Result obtained	additional info
null																				title required	no event created	The same with the other manda
	Duff																			location required	no event created	
		nul																		event created/the weather was not inserted	event created in the database; weather id reamain null; no weather undate for this event:	
			null																	etation date required	no event created	
																					no event created	
				null																ending date required	no event created	
					nuii															starting time required	no event created	
						null								J		J				ending time required	no event created	
							yes(any value)	null												Repeats is not 'no' but you did not chose an untill date !!	no event created	
									null											event created	event created normally; weather created if valid city name; null value assigned to description;	
																					event created normally; weather created if valid city name; null value assigned to pre	ecipitation and temperature;
			11/10/1990																	date before today	no event created	
			11/03/2015	10/03/2015																The ending date is before the starting date	no event created	
					4:00 am	1:00 am														The ending time is before the starting time	no event created	
			44/02/02/45		4.00 an	1.00 am		400000045												The well date is before the starting time	no even created	
			11/03/2015				yes(any value)	10/03/2015												The drift date is before the starting date	no event created	
																		yes	null	You haven't insert any User or these Users does not exist.	no event created	
																		yes	claudio;wiater;	You haven't insert any User or these Users does not exist	no event created	wrongly inserted wlater inste
														J		J		yes	claudio walter;	You haven't insert any User or these Users does not exist: walter claudio	no event created	missing ; between na
			11/03/2015	12/03/2015			yes(any value)													You can not repeat an event that last more than 1 day for now	no event created	
			11/03/2015	12/03/2015																There are some event in the middle of the one you want to create!	no event created	where was a event already created that would have been in conflict wit
		asdahsfshf																		event created	event created in the database; weather id reamain null; the system will try again to find weather during weath	city name not valid
																		VPS		event created	event created in the database: no invitation sent	empty username betwenn ":" w
						+												100	walter claudio	event created	quest created in the database invitation sent to walter a claudio	last username is accepted also
																					event created in the database. If the recognized or not real weather added to the database including early	mar datiriarie la accepted alar
							no	1		no	1	1	1	1	1					event created	(new invitation notification) to all invited users;	
																					event created in the database; if city recognized or not null weather added to the database; invitation sent	
							no			yes										event created	(new invitation notincation) to all invited users, bad conditions created in the database;	
																					first one: If city recognized or not null weather added to the database: invitation sent (new invitation	
							yes(any value)			yes										event created	notification) to all invited users for each event; bad conditions created in the database for each event;	
																					for each day an event created in the database with the same attribute (except for the date and time) of the first one: if city recognized or not null weather added to the database: invitation sent (new invitation	
			11/03/2015	12/03/2015			no	1												event created	notification) to all invited users for each event: bad conditions created in the database for each event:	I
																					event created in the database; if city recognized or not null weather added to the database; invitation sent	
																			walter, walter,	event created	(new invitation notification) to all invited users; bad conditions created in the database;	only one invitation sent, duplicate na
																			walter	event created	event created in the database; if city recognized or not null weather added to the database; invitation sent (new invitation notification) to all invited users; had conditions created in the database;	walter was the eventow