WeCalEvent

Testing Document

Software Engineering 2

Authors:

Mert Ergun

Mohamed Mehdi Kaabi

Nazrin Javadova

Milano – 2014

LaTeX

Contents

1 Introduction

1.1 System overview

WeCalEvent is a platform where users are able to create and manage their events and invite other users. Information about time and place of the event should be provided while creation of the event.

The system alerts the event creator 3 days before the event day in case of weather condition alteration and suggests him the closest day with suitable weather. Invited users will be able to decline or accept the invitation. All event participants will be notified in case of weather forecast alteration a day before the event.

This platform is sufficient for users who look forward to schedule their events according to the weather conditions.

1.2 Purpose of this document

The purpose of this acceptance test is to make sure that the project complies with the requirements of requirement documents.

1.3 Intended audience

The intended audience is:

- Team members

- Course staff

1.4 Scope

This document includes the plan, items, scope, approach, environment and procedure of WeCalEvent acceptance test. After that, the responsibilities of developers and user representatives are identified. At last, risks and contingencies are specified to ensure the test reliability. Other information not related to the test activities is not included in this document.

1.5 Definitions

Keyword	Definition
User	Is someone, who uses the application.
System	Our application

2 Testing

2.1 General approach

In order to test our project, we decided to create a manual test for each functionality and to use some automated tests. The performed tests will check the match between our application and the specifications written in requirements document.

2.2 Environment

we use test cases to be sure about our application stability. To perform this task we decide to use Junit. Its a simple Java tool that allow us to write an understandable and easy manner the unit tests.

2.3 Features to be tested

- Registration

- Login/Logout

- Create an event

- Update an event

- Delete an event

- Answer event invitation

- Search users

- Update profile

- Import/Export calendar

- Notify users

2.4 Features not to be tested

Search an event feature will not be tested as we decided not to have it in our application, as all events can be found while viewing calendars.

2.5   Test Case Report

Test Case ID	The ID or the number of Test Case
Test Case Description	Description of the test case
Actors Involved	The actors of this Test Case
Precondition	The state of the system at the time the test starts
Main path	List of steps that need to be applied for this Test Case
Expected result	The result that should be observed from a successful test.
Conclusion	- Failed - Successful

3 Test Cases

3.1 Registration

Test Case ID	TC 1.
Test Case Description	Registration with all fields filled in right
Actors Involved	System, User
Precondition	
Main path	- The user opens the web page. - The system shows the web page to user. - The user clicks on "Register" button. - The system shows the form page to user. - The user fills the form. - The user clicks on "submit" button.
Expected result	The user should be registered.
Conclusion	Successful

Test Case ID	TC 2.
Test Case Description	Registration with all or some fields not filled in
Actors Involved	System, User
Precondition	
Main path	- The user opens the web page. - The system shows the web page to user. - The user clicks on "Register" button. - The system shows the form page to user. - The user fills the form. - The user clicks on "submit" button.
Expected result	The user should not be registered and a notification about fields not filled in should appear.
Conclusion	Successful

Test Case ID	TC 3.
Test Case Description	Registration with already used email
Actors Involved	System, User
Precondition	
Main path	- The user opens the web page. - The system shows the web page to user. - The user clicks on "Register" button. - The system shows the form page to user. - The user fills the form. - The user clicks on "submit" button.
Expected result	The user should not be registered and a notification about used e-mail should appear.
Conclusion	Successful

Test Case ID	TC 4.
Test Case Description	Registration with wrong password confirmation
Actors Involved	System, User
Precondition	
Main path	- The user opens the web page. - The system shows the web page to user. - The user clicks on "Register" button. - The system shows the form page to user. - The user fills the form. - The user clicks on "submit" button.
Expected result	The user should not be registered and a notification about wrong password confirmation should appear.
Conclusion	Successful

Test Case ID	TC 5.
Test Case Description	Making calendar public while registration
Actors Involved	System, User
Preconditon	
Main path	- The user opens the web page. - The system shows the web page to user. - The user clicks on "Register" button. - The system shows the form page to user. - The user fills the form making calendar public. - The user clicks on "submit" button.
Expected result	The user should be registered and other users should be able to see events' information in his/her calendar
Conclusion	Successful

Test Case ID	TC6.
Test Case Description	Making calendar private while registration
Actors Involved	System, User
Preconditon	
Main path	- The user opens the web page. - The system shows the web page to user. - The user clicks on "Register" button. - The system shows the form page to user. - The user fills the form making calendar private. - The user clicks on "submit" button.
Expected result	The user should be registered and other users should not be able to see events' information in his/her calendar
Conclusion	Successful

3.2 Login/Logout

Test Case ID	TC 7.
Test Case Description	Logging with all fields filled in right
Actors Involved	System, User
Precondition	The user must be registered
Main Path	• The user opens the web page. • The system shows the home page to user. • The user enters his email and password in the input form. • The user clicks on "Login" button.
Expected result	The user should be logged in
Conclusion	Successful

Test Case ID	TC 8.
Test Case Description	Logging with all or some fields not filled in
Actors Involved	System, User
Precondition	The user must be registered
Main Path	• The user opens the web page. • The system shows the home page to user. • The user fills the input form. • The user clicks on "Login" button.
Expected result	The user should not be logged in
Conclusion	Successful

Test Case ID	TC 9.
Test Case Description	Logging in with wrong e-mail
Actors Involved	System, User
Precondition	The user must be registered
Main Path	• The user opens the web page. • The system shows the home page to user. • The user enters his email and password in the input form. • The user clicks on "Login" button.
Expected result	The user should not be logged in
Conclusion	Successful

Test Case ID	TC 10.
Test Case Description	Logging in with wrong password
Actors Involved	System, User
Precondition	The user must be registered
Main Path	• The user opens the web page. • The system shows the home page to user. • The user enters his email and password in the input form. • The user clicks on "Login" button.
Expected result	The user should not be logged in
Conclusion	Sucessful

Test Case ID	TC 11.
Test Case Description	Logging out
Actors Involved	System, User
Precondition	• The user must be registered • The user must be logged in.
Main Path	The user clicks on the logout button.
Expected result	The user should be logged out.
Conclusion	Sucessful

3.3 Create an event

Test Case ID	TC 12.
Test Case Description	Creating event with all fields filled in right
Actors involved	System, User
Precondition	The user must be logged in
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should be created and appear in the calendar.
Conclusion	Successful

Test Case ID	TC 13.
Test Case Description	Creating event with all or some fields not filled in
Actors involved	System, User
Precondition	The user must be logged in
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should not be created and a notification about fields not filled in should appear
Conclusion	Successful

Test Case ID	TC 14.
Test Case Description	The event starts when another event didn't finish yet
Actors involved	System, User
Precondition	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should not be create and a notification about conflict with another event's time should appear.
Conclusion	Successful

Test Case ID	TC 15.
Test Case Description	The event ends after another event stats
Actors involved	System, User
Precondition	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should not be create and a notification about conflict with another event's time should appear.
Conclusion	Successful

Test Case ID	TC 16.
Test Case Description	Two events start and end at the same time
Actors involved	System, User
Precondition	The user must be logged in.
Main path	• The users clicks on "create an event" button. • The system loads event creation page. • The user fills the fields with event information. • The user clicks on "create" button.
Expected result	The event should not be create and a notification about conflict with another event's time should appear.
Conclusion	Successful

Test Case ID	TC 17.
	The event starts and ends when another event is going
Actors involved	System, User
Precondition	The user must be logged in.
Main path	• The users clicks on "create an event" button. • The system loads event creation page. • The user fills the fields with event information. • The user clicks on "create" button.
Expected result	The event should not be create and a notification about conflict with another event's time should appear.
Conclusion	Successful

Test Case ID	TC 18.
	Creating event with wrong location name
Actors involved	System, User
Precondition	The user must be logged in
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should not be created and a notification about wrong location name should appear.
Conclusion	Succesfull

Test Case ID	TC 19.
Test Case Description	Creating event with right location name
Actors involved	System, User
Precondition	The user must be logged in
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the location field.
Expected result	The system checks for the location and a notification about right location name should appear.
Conclusion	Succesfull

Test Case ID	TC 20.
Test Case Description	The event ends before it starts
Actors involved	System, User
Precondition	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should not be created.
Conclusion	Successful

Test Case ID	TC 21.
Test Case Description	Creating public event
Actors involved	System, User
Precondition	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information making the event public. - The user clicks on "create" button.
Expected result	The event should be public.
Conclusion	Successful

Test Case ID	TC 22.
Test Case Description	Creating private event
Actors involved	System, User
Precondition	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information making the event private. - The user clicks on "create" button.
Expected result	The event should be private
Conclusion	Successful

Test Case ID	TC 23.
Test Case Description	Inviting users while creating an event
Actors involved	System, User
Precondition	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with right event information inviting other users. - The user clicks on "create" button.
Expected result	The event should be created and invited users should get an invitation notification
Conclusion	Successful

Test Case ID	TC 24.
Test Case Description	Creating an event in past
ActorB Involed	System, User
Preconditon	The user must be logged in.
Main path	- The users clicks on "create an event" button. - The system loads event creation page. - The user fills the fields with event information. - The user clicks on "create" button.
Expected result	The event should not be created and notification that event can't be created should appear.
Conclusion	Successful

3.4   Update an event

Test Case ID	TC 25.
Test Case Description	Updating event with all fields filled in right
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created.
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Update Event" button. - The event creator updates the event information. - The user clicks on "Update" button.
Expected result	The event should be updated and its information in calendar should change.
Conclusion	Successful

Test Case ID	TC 26.
Test Case Description	Updating event with wrong location name
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created.
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Update Event" button. - The event creator updates the event information. - The user clicks on "Update" button.
Expected result	The event should not be created and a notification about wrong location name should appear.
Conclusion	Successful

Test Case ID	TC 27.
Test Case Description	Updating event with right location name
Actors Involved	System, User
Precondition	The user must be logged in
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Update Event" button. - The event creator updates the event information. - The user clicks on "Update" button.
Expected result	The system checks for the location and a notification about right location name should appear.
Conclusion	Successful

Test Case ID	TC 28.
Test Case Description	The event ends before it starts
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created.
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Update Event" button. - The event creator updates the event information. - The user clicks on "Update" button.
Expected result	The event should not be updated.
Conclusion	Successful

Test Case ID	TC 29.
Test Case Description	The event starts when another event didn't finish yet
Actors Involved	System, User
Precondition	• The user must be logged in. • The event must be created.
Main path	• The event creator clicks on an event. • The system loads event view page with existing information. • The user clicks on "Update Event" button. • The event creator updates the event information. • The user clicks on "Update" button.
Expected result	The event should not be updated.
Conclusion	Successful

Test Case ID	TC 30.
Test Case Description	The event ends after another event stats
Actors Involved	System, User
Precondition	• The user must be logged in. • The event must be created.
Main path	• The event creator clicks on an event. • The system loads event view page with existing information. • The user clicks on "Update Event" button. • The event creator updates the event information. • The user clicks on "Update" button.
Expected result	The event should not be updated.
Conclusion	Successful

Test Case ID	TC 31.
Test Case Description	Two events start and end at the same time
Actors Involved	System, User
Precondition	• The user must be logged in. • The event must be created.
Main path	• The event creator clicks on an event. • The system loads event view page with existing information. • The user clicks on "Update Event" button. • The event creator updates the event information. • The user clicks on "Update" button.
Expected result	The event should not be updated.
Conclusion	Successful

Test Case ID	TC 32.
Test Case Description	The event starts and ends when another event is going
Actors Involved	System, User
Precondition	• The user must be logged in. • The event must be created.
Main path	• The event creator clicks on an event. • The system loads event view page with existing information. • The user clicks on "Update Event" button. • The event creator updates the event information. • The user clicks on "Update" button.
Expected result	The event should not be updated.
Conclusion	Successful

Test Case ID	TC 33.
Test Case Description	The event starts in the past
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created.
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Update Event" button. - The event creator updates the event information. - The user clicks on "Update" button.
Expected result	The event should not be updated and notification that event can't be created should appear.
Conclusion	Successful

Test Case ID	TC 34.
Test Case Description	Updating event with some or all fields not filled in
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created.
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Update Event" button. - The event creator updates the event information. - The user clicks on "Update" button.
Expected result	The event should not be updated and notification that event can't be created should appear.
Conclusion	Successful

Test Case ID	TC 35.
Test Case Description	Inviting user while Updating an event
Actors Involved	System, User
Precondition	• The user must be logged in. • The event must be created.
Main path	• The event creator clicks on an event. • The system loads event view page with existing information. • The user clicks on "Update Event" button. • The event creator updates the event information. • The user clicks on "Update" button.
Expected result	The event should be updated and invited user should get a notification about invitation.
Conclusion	Successful

3.5 Delete an event

Test Case ID	TC 36.
Test Case Description	Deleting an event
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created.
Main path	- The event creator clicks on an event. - The system loads event view page with existing information. - The user clicks on "Delete" button.
Expected result	The event should be Updated and its information in calendar should change.
Conclusion	Successful

3.6 Answer event invitation

Test Case ID	TC 37.
Test Case Description	Accepting event invitation
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created. - The user must be invited.
Main path	The user accepts to participate in event
Expected result	The event should be accepted, should appear in user's calendar and user should appear in participants list.
Conclusion	

Test Case ID	TC 38.
Test Case Description	Declining event invitation
Actors Involved	System, User
Precondition	- The user must be logged in. - The event must be created. - The user must be invited.
Main path	The user declines to participate in event
Expected result	The event should be declined.
Conclusion	

3.7 Search users

Test Case ID	TC 39.
Test Case Description	Users search
ActorsB Involved	System, User
Precondition	The user must be logged in.
Main path	• User types needed user's name in "search user" field
Expected result	Needed user's calendar should open
Conclusion	Successful

3.8 Update profile

Test Case ID	TC 40.
Test Case Description	Updating profile with all fields filled in right
ActorsB Involved	System, User
Precondition	The user must be logged in.
Main path	- The event creator clicks on "Update profile". - The system loads personal information page with existing information. - The user updates his information. - The user clicks on "update" button.
Expected result	Profile should be updated. Updated fields should be changed
Conclusion	Successful

Test Case ID	TC 41.
Test Case Description	Updating profile by replacing e-mail by used one
ActorsB Involved	System, User
Precondition	The user must be logged in.
Main path	- The event creator clicks on "Update profile". - The system loads personal information page with existing information. - The user updates his information. - The user clicks on "update" button.
Expected result	Profile should not be updated and notification about used e-mail should appear.
Conclusion	Successful

Test Case ID	TC 42.
Test Case Description	Updating profile with wrong password confirmation
ActorsB Involved	System, User
Precondition	The user must be logged in.
Main path	• The event creator clicks on "Update profile". • The system loads personal information page with existing information. • The user updates his information. • The user clicks on "update" button.
Expected result	Profile should not be updated and notification about wrong password confirmation should appear.
Conclusion	Successful

Test Case ID	TC 43.
Test Case Description	Updating profile with all or some fields not filled in
ActorsB Involved	System, User
Precondition	The user must be logged in.
Main path	• The event creator clicks on "Update profile". • The system loads personal information page with existing information. • The user updates his information. • The user clicks on "update" button.
Expected result	Profile should not be updated and a notifacation about fields not filled in should appear.
Conclusion	Successful

3.9 Import/Export calendar

Test Case ID	TC 44.
Test Case Description	Import calendar
Actors Involved	System, User
Precondition	The user must be logged in.
Main path	- The user clicks on ex/import button. - The system loads export/import page with existing information. - The user clicks on . - The user .
Expected result .	The calendar should
Conclusion	

Test Case ID	TC 45.
Test Case Description	Export calendar
Actors Involved	System, User
Precondition	The user must be logged in.
Main path	- The user clicks on ex/import button. - The system loads export/import page with existing information. - The user clicks on export button.
Expected result	The calendar should be expored.
Conclusion	Successful

3.10   Notify users

Test Case ID	TC 46.
Test Case Description	Notify participant each 12 hours about the weather forecast
Actors Involved	System, User
Precondition	• The event must be created. • The user must be participant of this event.
Main path	• Weather forecast changed. • System sends a notification to all participants' home pages. • System sends e-mails to each participant to concerned event.
Expected result	E-mail and notification sent out.
Conclusion	Successful

Test Case ID	TC 47.
Test Case Description	Notify the participant users one day before the event starting date if the forecast change
Actors Involved	System, User
Precondition	• The event must be created. • The user must be participant of this event.
Main path	• Weather forecast changed. • System sends a notification to all participants' home pages. • System sends e-mails to each participant to concerned event.
Expected result	E-mail and notification sent out.
Conclusion	Successful

Test Case ID	TC 48.
Test Case Description	Notify event creator three days before event starting date if the forecast change
Actors Involved	System, User
Precondition	• The event must be created. • The user must be a creator of this event.
Main path	• Weather forecast changed. • System sends a notification to the creator and suggest him the closet date with the desired weather. • System sends e-mails to to the creator and suggest him the closet date with the desired weather.
Expected result	E-mail and notification sent out.
Conclusion	Successful