Test results

Test case 1 - JUnit test

Unit tests are all green - Passed

Test case 2 - Integration test

The timelines are created and represented correctly. Even the exceptional cases are handled properly. - Passed.

Test case 3 -JUnit test

Unit tests show all green. -Passed.

Test case 4 - Integration test

Events are created and shown correctly. Multiple events on same day is also handled as intended. Exceptional cases are handled. -Passed

Test case 5 - Integration test

Durational events are created and shown as a bar (as intended). Multiple durational events are also shown properly. Exceptional cases are handled with the expected results. -Passed

Test case 6 - JUnit test

This test is found in the EventTest file, and shows all green. -Passed.

Test case 7 - Integration test

The delete function is working properly. Timelines are deleted as intended. -Passed.

Test case 8 - Integration test

The delete function for events is working as intended. Events are removed. - Passed.

Test case 9 - Integration test

The exporting of timeline files are working as intended. User can select file location, and the file is saved properly. -Passed.

Test case 10 - Integration test

Loading of timelines are working as intended. User can load timelines previously exported in this application and all the events linked to the timeline are shown properly.

Software testing report

Create mode

The create mode is able to create one or more timelines, add any number of events and create events with and without duration. The create mode also have the modify function for both events and timelines, and also the delete function for both events and timelines. All the create mode requirements have been met.

Display mode

The display mode handles all the visual representations of timeline and/or events. The display mode can correctly display loaded or created timelines with the events linked to selected timeline. The software can properly handle displaying multiple events both with and without duration. The events are shown in correct chronological order, and the event details are shown properly in another window.

The application does not show multiple timelines simultaneously, but are shown and stored in “my projects” tab, where the user can select which timeline he/she wants to view. This was a deliberate design decision which makes the timeline handling smoother, since showing multiple timelines in the same window will be messy.
All the display requirements have been met.

General Requirements

The application have a functional save and load function, where the application can export the files as an .xml file as well as load and display them correctly. The general requirements has been met

Delivery testing

Tools

The delivery testing was made using a virtual machine running Microsoft Windows 7, MacOS and Linux Ubuntu v16. We used VirtualBox from Oracle as our virtual machine.

Windows 7

Without installing Java, the jar file could not be run, and the operating system could not handle the file. After installing java however, the application ran very smoothly. All the functions were manually tested and all non-unit tests were run successfully.

Linux Ubuntu

Without installing Java, the application cannot be run, you only open the file as an archive. After right clicking the jar and adding the permission “allow running as executable”, application worked smoothly. All non-JUnit tests were run properly and with expected results.

MacOS - Mac OS Sierra 10.12.5

As the previous operating systems, the system cannot handle the files unless Java is installed. After installing java, the application starts up as normally. All non-JUnit tests were run properly with expected results.

Results:

We agreed to have a readme file in the delivery which informs the user that Java is a prerequisite and needs to be installed, before this application can be executed. The ‘readme’ file will also have the address to https://www.java.com.

The system is behaving as expected and is ready for delivery.