Screenshot Comparison Tests

Enabling the Screenshot Comparison Tests

The bundled screenshot comparison tests are disabled by default, because differing browser versions or operating systems may cause the tests to fail. The tests can be found in the `com.vaadin.testbenchexample.ScreenshotITCase` class and can enabled by removing the `@Ignore` annotation on the `ScreenshotITCase` class declaration.

Running the Tests

Once the screenshot comparison tests have been enabled, the tests can be run as described in the README file by invoking `mvn install`, just like all the other tests.

Screenshot Files

There are two kinds of screenshot files: reference files and error files. The reference files are the images that the screenshots grabbed during the test run are compared against. Reference files should be stored in the reference directory. If a comparison fails, the differing screenshot along with a HTML file displaying the differences are stored as error files in the error directory.

The example project is configured so that the reference files are stored in `src/test/resources/screenshots` and any eventual error files are stored in `target/testbench/screenshot_errors`. See `ScreenshotITCase.setUp()` for details on how this is done.

Updating the Reference Images

The screenshot references must be updated if the screenshot comparison tests fail. This might be due to a different browser or OS version, which renders fonts or native input elements differently than the existing screenshot references.

When a screenshot comparison fails, and thus fails the test, a screenshot is written to the error directory. This happens if the comparison notices differences between the reference image and the new screenshot, OR if there is no reference image available.

Updating Normal Reference Images
The simplest case of updating the reference image is to just move the PNG file from the error directory (`target/testbench/screenshot_errors`) to the reference directory (`src/test/resources/screenshots`). This works perfectly for the `ScreenshotITCase.testOnePlusTwo()`, which produces a file named `oneplustwo_<OS>_<browser>_<browser verson>.png`, e.g.

`oneplustwo_mac_firefox_20.png`. After inspecting the HTML report found in the error directory to make sure that there are no functional regressions and moving the new screenshot to the reference directory, the test will pass again.

Updating Masked Reference Images

There is also another screenshot test case, which uses masked screen shots. This is the `ScreenshotITCase.testOnePlusTwoWithRandomLog()` test case. Updating this screenshot is a bit more involved, as you must mask out the region containing the calculation logs in some kind of image editing software. The basics are the same as for normal reference images: Copy the error screenshot to the reference directory. After the screenshot is copied, mask out the calculation log table.

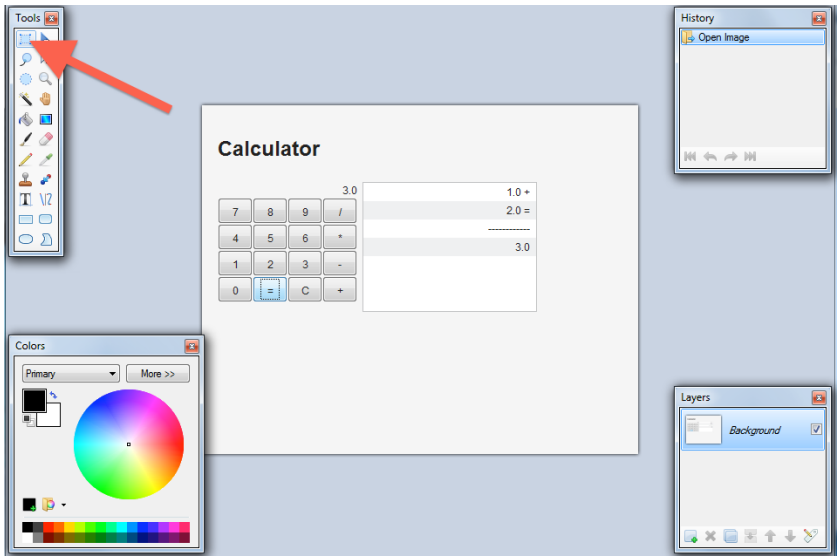
Windows: A suitable free image editor is paint.net, which you can download from http://getpaint.net. Navigate the site carefully, as it has some pretty confusing ads. Photoshop or some other advanced image editor would also suffice and the instructions should be applicable.

Mac: Mac OS X already comes bundled with Preview.app, which is sufficient to perform the steps listed below.

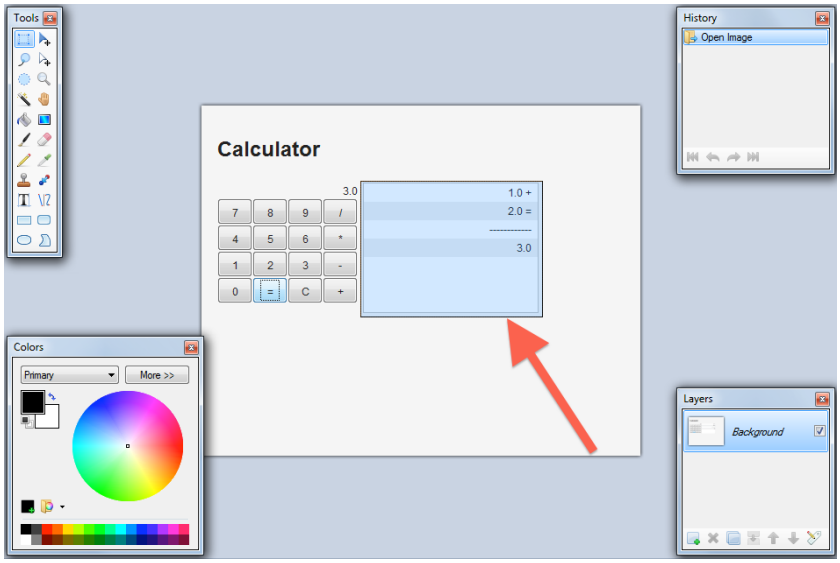
Linux: Gimp (http://www.gimp.org) is a good choice for any Linux distribution.

Follow these steps to mask the screenshot:

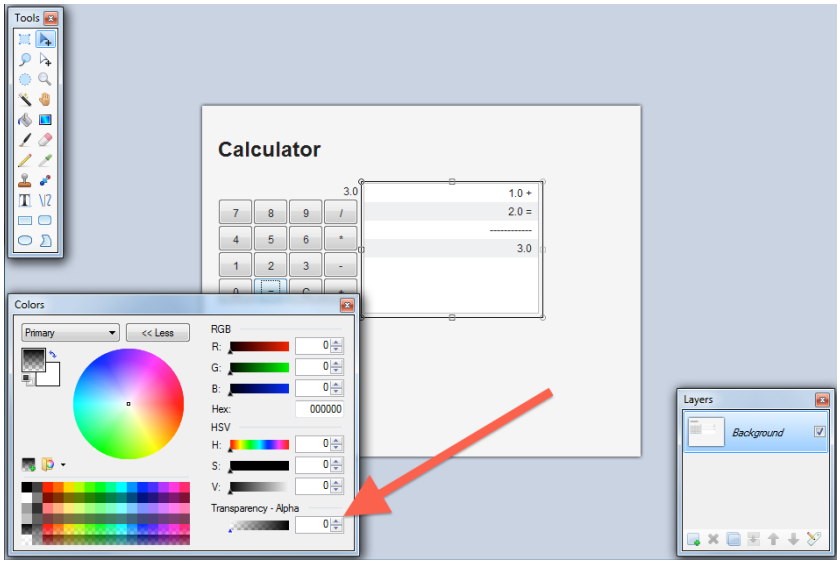
1. Open the screenshot in the editor (paint.net, Preview, or Gimp)



2. Choose the rectangular selection tool, this is selected by default in Preview on the Mac.

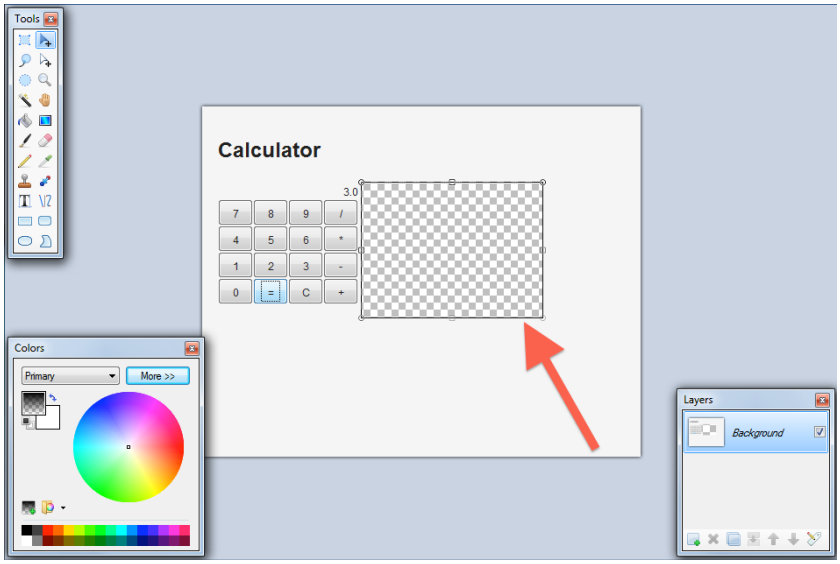


3. Select the area containing the calculation log
4. **On Windows:** Click the “More >>” button in the color picking window and change



the alpha (transparency) value to 0. This step is not necessary in Preview.

5. Press the Delete button on your keyboard. The selected area will be replaced with



a transparent color.

6. Save the image as the new reference image.

The tests should now pass.