

Assignment 1

CSE 4313 3.0 Software Engineering Testing, Winter 2013, Section M

Due: Monday, January 28, 2013, 2:30pm.

Format: Individual.

Editing and Evaluating bugs

The purpose of this assignment is to give you experience reporting bugs and developing test cases. This task will give you practice thinking how to develop a test case from a requirements document, what a professional bug report should be, as well as analyzing bugs before reporting them.

To get started

- Download BORG Calendar installation package (BORG_installer-1.7.7.jar) (Link 1 in course website) from <http://sourceforge.net/projects/borg-calendar/?source=navbar>. You should run installation package by typing `java -jar BORG_installer-1.7.7.jar` and follow screen instructions. Remember your installation path defined in step 4 (The default installation path is `$HOME/BORGCalendar`). After product deployment, you should change current directory to the BORG installation path and then run BORG Calendar by typing `./run.borg.sh` to get familiar with the tool.
- Download BORG Calendar product specification (Link 2 in course website) from <http://borg-calendar.svn.sourceforge.net/viewvc/borg-calendar/trunk/BORGCalendar/res/requirements/SRS%20for%20BORG%20Calendar.pdf?revision=1431> and explore calendar developer website (Link 3 in course website) (<http://mbcsoft.com/>)
- Join the QA project of BORG Calendar at (Link 4 in course website)
<https://sourceforge.net/user/registration>
The user name must start with your last name, and you must use your cs email account.

What to do

Submit 5 reports, either reporting bugs or proposing enhancements. For each report focus on the following:

- Clarity and tone.

- Attempt to replicate the bug.
- Perform further analysis using various types of follow-up testing.
- Include important information that you successfully replicated the bug on XXX configuration in YYY build, or describing a simpler set of replication steps (if you found one). Your comments should NEVER appear critical or disrespectful of the developer.

Issues to consider when you working on bug report:

- Is the summary short (about 50-70 characters) and descriptive?
- Can a developer understand the bug report? Is it sufficient details to envision what the program did in response? Is it clear what the failure was?
- Is it obvious where to start (what state to bring the program to) to replicate the bug?
- Is it obvious what files to use (if any)? Is it obvious what you would type?
- Is the replication sequence provided as a numbered set of steps, which tell developer exactly what to do and, when useful, what developer will see?
- Does your bug report include unnecessary information, personal opinions or anecdotes that seem out of place?
- Is the bug report too long? Too short? Does it have a lot of unnecessary steps?
- Is it overly general, e.g. "Insert a file and you will see"? Is there an example, like "Insert a file like `blah.foo` or `blah2.fee`"?
- Can you replicate the bug by following your step?
- Can developer get lost or wonder whether you had done a step correctly? Would additional feedback, e.g. "the program will respond like this...", have helped?
- Does configuration or environment change effect on bug reproduction?
- What were the follow-up tests that required to run on this bug report?
- Does the description include statements about why this bug would be important to the customer or to someone else? The bug report need not include such information, but if it does, it should be credible, accurate, and useful.

Create a written report that includes the report IDs for the 5 reports, an explanation of all reports, steps that you used to discover bugs and at least one test case scenarios per bug. Also explain what functionality you tested and include relevant screen shots. Since you will probably NOT have time or the resources to run many follow-up tests, describe other tests which you would

like to run. You should provide justification of these follow-up tests and explain what would you hope to learn from them. If you propose new functionality, then justify your features request. For this, you may include pictures or drawings.

What to Submit

Submit a hard copy of your written report to the 4313 drop box by the deadline. Submit the same report electronically before the deadline as well (PDF format is preferable, but other formats will also be accepted). To submit, give:

```
submit 4313 a1 a1.pdf
```

where `a1.pdf` is your report.

The marker will mark the hard copy submission. Electronic submissions will be used for re-marking purposes.

Grading

This assignment will be marked out of 100, 20 points per bug report. For each bug, 10 marks will relate to the reported bug you added online, and 10 to the written report.

The content of your online bug report can vary depending on the problem. The key thing is that they are useful to the reader. Following is a list of things that can give you points:

- Reporting the configuration and build you used.
- Reporting a simpler set of replication steps.
- Reporting the results of a **good** follow-up test.
- Making a good argument regarding importance (pro or con).

For your written report, the marker will be looking for details that suggest you had insight into what makes a good test case scenario. How well you justify your feature request and how important this functionality for this product.