SE420, SQA: Exercise #5 – System/Sub-system Prototype and Regression Testing

GROUP SQA Assignment

DUE: As Indicated on Canvas

Please thoroughly read SQA Chapter 8 and 9. Please download GitHub Desktop for Windows or Mac and check-out Examples-RAID-Unit-Test, Examples-Imageproc-Unit-Test and Examples-Crypto-Unit-Test (checked in by siewertserau) to GitHub public Git repository for source code CMVC and use it to update your code with changes that are tracked with CMVC and comments that indicate what was changed, why, and if it fixes a defect from Exercise #4 and how that was verified. Transfer all your bugs into PRClab Bugzilla (http://prclab.pr.erau.edu/) and set up your account if you need to.

You may work on this project in a group of 2 or 3 students on a common software module selected in Assignment #4. Please clearly identify all group members on the title page of your submission and indicate what sections of the overall effort each team member worked on [e.g. Analysis, Design, Prototype coding OR on the right-hand side of the V, test plans, design and testing of tests, execution of tests].

The goal of this lab is to gain experience fixing defects identified in code modules found through your own debugging in earlier exercises and to further improve your proof-of-concept “File RAID”, “Image Processing Pipeline” or “Message Crypto” software or other open source improvement project of your interest. It is expected that you may have requirements changes, design changes, code restructuring and refactoring, defects to fix, and new testing to complete. You are expected to do so using Git to track and justify your changes to the base you created in Assignment #4 and Bugzilla to document your self-reported bugs, categorized as defects, deficiencies, or enhancements and provide evidence that code has been well tested at the unit level, integration level, and system level as you go. For this assignment the focus should be unit level regression testing and further development of integration and system tests.

Exercise #5 Requirements:

1) [25 points] Create a Bugzilla account for yourself on http://prclab.pr.erau.edu/ and enter all bugs you know of from your Assignment #4 and categorize them as defects, deficiencies, or enhancements (or other) and when you’re done, provide a report for your product with all bugs listed [You should have at least 10] and provide the list in a prioritized order of what should be done first and what should be done last. If you are a team, you should have N times the number of bugs and should assign them to members of your team to fix, design, and/or code. Spend time on this and make high quality entries that I can browse when I grade. If you would like changes made to your product, format or fields used in bug tracking, please send requests to siewerts@erau.edu and be clear about what you’d like.

2) [25 points] Update your proof-of-concept product requirements based on Assignment #3 and #4 feedback and your improved group understanding by merging the best requirements determined in you individual efforts:
   a) Define the product better:
      i) [15 points] Re-write specific requirements, updated, based on what you originally wrote in Assignment #3, what you were asked to do in Assignment #4, and any feedback you get from your instructor. Your goal should be to simplify and clarify, but to also not be vague and provide requirements that can be validated and used in testing for verification.
      ii) [10 points] Update your acceptance test outline or stubbed out C code based on requirements updates and make sure that each acceptance test case still maps to a specific requirement.

3) [25 points] Take a first cut at documenting your design at system and architecture levels using block diagrams, data flow diagrams, and simple state machines or logic flowcharts [based on your experience] to better define your proof-of-concept product:
   a) Define and refine the product design:
      i) [15 points] Provide a single page high-level block diagram that clearly identifies key modules of you design and interfaces each has with other modules in your overall applications.
      ii) [10 points] Write English descriptions of what each module (block) in your high-level design must do (per requirements) and for one or more of them, further detail the design to show functions, shared buffers, argument vectors, messages or other types of interfacing and communication between your modules.

4) [25 points] Improve your proof-of-concept code by either restructuring to meet refined requirements or design work from previous steps and/or to fix specific defects, deficiencies, or enhancements you have identified at a code level:
   a) Define and refine the C code implementation:
      i) [15 points] Provide a second version of each one of your modules previously identified, add any new modules and restructure or refactor any code as needed to improve quality, address new requirements, design changes, the bug list, or to improve the design of your proof-of-concept product.
      ii) [10 points] Test code modifications, creating new test code as needed and check in all code updates including new test code into your Git repository with good commenting on the reason for change.

Overall, provide a well-documented professional report of your findings, output, and tests so that it is easy for a colleague (or instructor) to understand what you've done, what worked, what did not and why (even if you can't complete to your satisfaction). Include any C/C++ source code you write (or modify) and Makefiles needed to build your code. I will look at your report first, so it must be well written and clearly address each problem providing clear and concise responses to receive credit, but I will look at your code and test results as well if I have questions.

In this class, you’ll be expected to consult the Linux manual pages and to do some reading and research on your own, so practice this in this first lab and try to answer as many of your own questions as possible, but do come to office hours and ask for help if you get stuck.

Upload all code and your report completed using MS Word or as a PDF to Blackboard and include all source code (ideally example output should be integrated into the report directly, but if not, clearly label in the report and by filename if test and example output is not pasted directly into the report). ***Your code must include a Makefile so I can build your solution on PRClab. Please zip your solution with your last name embedded in the file name.***

Grading Rubric

[25 points] Bug reporting and classification of defects, deficiencies, and enhancements to code, design and requirements:

[10 pts] Number of bugs and completeness____

[10 pts] Quality and clarity of reporting____

[5 pts] Classification_________________________________ __________________________

[25 points] Define and refine requirements [evolution #2]:

[15 pts] New and improved requirements____

[10 pts] Acceptance test updates____

[25 points] Define and refine high-level design and some detail [evolution #2]:

[15 pts] High level design of PoC____

[10 pts] Select detailed design work____

[25 points] Define and refine requirements [evolution #2]:

[15 pts] New and improved PoC code updates_____________________________________

[10 pts] Testing and CMVC tracking____