SE420, SQA: Exercise #6 – Product Completion and Acceptance Testing

GROUP SQA Assignment

DUE: As indicated on Canvas

The goal of this assignment is to complete the application you selected to work on individually or in a group of 2-3 students as specified in assignment #4 by updating your requirements, refining your design, fixing defects you have found in your implementation, re-integrating your software modules and providing a final set of tests for regression and acceptance testing of your final product.

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].

During our Final Oral Exam in SQA, each group will be asked to host an inspection of their sub-system design and prototype software modules including requirements review, architecture design walk-through, and software module walk-throughs. If the effort is a group effort, each group member should have a clear role on the team (specific contributions) and all team member will be expected to present requirements, design and source code.

Exercise #6 Requirements:

1) [25 points] Create a final disposition for all bugs you have reported in Bugzilla by creating fixes to defects, de-scoping features, or classifying bugs as anomalies or release limitations noted in your final release notes. Close all bugs and provide a final summary report of them.

2) [25 points] List your final revised requirements and update your acceptance test:
   a) Define the product better:
      i) [15 points] Re-write specific requirements, updated, based on what you have learned and improved in your application.
      ii) [10 points] Update your acceptance test outline and complete as a C code driver or scripted execution of multiple C code drivers that demonstrates all major requirements for your application.

3) [25 points] Create a final report describing your application design:
      i) [15 points] Provide a final updated 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] Provide final 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] Provide your final integrated C code implementation with documentation of the unit tests and any regression tests you devised. Provide basic instructions on how to build, test and use your application.

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 of bug closure work ____

[5 pts] Closure of all bugs__________________________ _________________________

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

[15 pts] Final requirements____

[10 pts] Acceptance test updates____

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

[15 pts] High level design description____

[10 pts] Summary detailed design work____

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

[15 pts] Final code, build, run, test____

[10 pts] Useability of application____