

Final Project: Quality Assurance Plan

17-654: Analysis of Software Artifacts
Jonathan Aldrich (jonathan.aldrich@cs.cmu.edu)

Initial QA plan due Thursday, May 3 at 10:30am
QA plan review documents due May 8, 9am
QA plan presentations on May 8, 9-12am
Revised QA plan due Monday, May 15 at 10:30am
200 points total
This is an Master's (17-654) only assignment

The goal of the quality assurance plan assignment is to provide MSE/MSIT students with a practical opportunity to plan a comprehensive quality assurance program for their studio, practicum, or other project, and to evaluate QA plans written by peers.

The expected scope of effort for this project is around 18 hours per student; about 12 hours in developing the initial plan and your presentation, and 6 hours reviewing other plans and revising your plan based on reviews and feedback from the initial presentation. Late days may not be used.

Groups. Students may work on this project in groups of up to 6. Group projects will be given a single grade. You are free to choose your own groups, and should generally form groups that correspond to existing studio or practicum projects. Students not part of a studio or practicum group may join another group, or develop a QA plan for some other current, past, or future project in which they are involved. Expectations for the level of depth in the plan will be scaled to the size of the group; as a rule of thumb, your QA plan should be 2-3 pages (12-point single spaced) per group member.

Collaboration Policy. Since different groups will be working on different projects, the only collaboration policy is that your work must be your own (as always).

1 Initial Quality Assurance Plan (60 points)

Your quality assurance plan is a comprehensive description of how you intend to assure a quality product from your project. It generalizes the usual notion of a test plan to describe strategies for using reviews, static analysis and possibly other techniques in addition to testing.

You may choose any appropriate organization for your quality assurance plan document (if you are stuck, you might consider standards such as IEEE 730-2002, but this is not required). However, the document must discuss all of the following:

Header and Versioning. Your QA plan should include the name of your group or project and the names of the group members. It should include a version number and date. There should be a section describing the changes made to the document at each revision (this should cover at least the changes between the initial QA plan handed in and the final QA plan).

Context. Explain enough of the context of your project for reviewers to understand what you are doing and judge if the quality assurance plan is appropriate. This section may be a short summary combined with a pointer to more information (which should be bundled with your QA plan). Define at a high level what quality means to your team, and how you will measure it in your product.

Goals. Define more specifically the quality goals you intend to achieve. The goals should be clearly defined, measurable, and quantitative where possible (e.g. %, # defects/kLOC, limits on # clients for stress testing, etc.). Cover both functional requirements and quality attribute requirements. Describe the scope of your plan; what will you assure, and what will you *not* assure?

Strategy. Provide a 1-paragraph high-level summary of your quality assurance strategy, similar to the example summaries discussed in class (from the Kaner et al. book).

What tools and techniques will you use to assure the quality of each functional requirement and each quality attribute requirement? Consider static analysis, testing, reviews, and other techniques if applicable. Different approaches are likely to be useful for different quality attributes. You may find you want to assure a quality attribute using more than one approach. In this case, explain how the different approaches you chose complement one another.

In this QA document, you must include the reasons behind the strategic choices you make. This kind of rationale is not always part of QA plans in industry, but is important for the learning goals of this course. You should not feel you have to use every technique, but you should justify why you chose the technique you did and why you did not choose other techniques.

If you chose static analysis, why? Which tool will you use, or (perhaps) will you develop a new analysis? What benefits will this give you over other techniques?

Note: although the instructor has promoted static analysis techniques in this class, you may have good reasons for not including static analysis—e.g. cost of tools (if so, be specific about why they are too costly for their benefits) or lack of tools applicable to what you are doing. This is fine as long as you show that you have considered static analysis along with other techniques in coming to your final decision about what goes into the plan.

Every plan will probably include some kind of testing. What approaches will you use for testing—e.g., white box, black box, unit, acceptance, equivalence class testing? Are there design considerations that will enable testing, such as test scaffolding? How will you approach integration testing and/or regression testing? What is your criteria for completeness of testing?

What documents or parts of the code will you review? What checklists will you use for each relevant document, and for the code? What kind of review will be done in each case (note this may vary for different documents or for different sections of the code)? What process will you use in the reviews—e.g. will you tailor or leave out some of the process elements of the formal technical reviews discussed in class, and if so, why?

How will you assure quality of artifacts other than code?

Organization. What is the total amount of resources available for your project, and what proportion do you intend to devote to quality assurance? How will you organize your quality assurance team? (Note: the complexity of your answer will of course depend on team size)

How will you manage artifacts related to quality assurance, such as version control for test cases, a test automation framework, documentation of tests, review results, and bug databases? What process will you use to ensure that defects found via testing, reviews, or static analysis are properly prioritized and addressed?

How does your QA strategy break down into tasks, how much effort will you devote to each task? When will each task begin and end, i.e. what is the schedule for QA tasks? For each task, what is the exit criterion (i.e. how will

you know when you are done?) and what are the deliverables? Note that tasks like testing may be integrated into phases like coding, especially for unit tests in teams using agile processes. If you are using an agile process, you may not have a dedicated test team, but you should still estimate the amount of resources (e.g. person-hours) you will be using to support various aspects of quality assurance on your project.

What will you measure about your quality assurance processes (this may be different for e.g. testing and reviews), and what approach will you use to improve your QA processes as you go along?

References. Throughout your quality assurance plan, refer to other relevant documentation (if applicable) such as requirements, design, or architecture documents for your project. Turn in copies of these documents for review along with your quality assurance plan, as a zip file, to Blackboard. If a particular document would be useful to your reviewers but has not been written yet, say so and provide substitute information if possible.

Grading. The quality assurance plan will be graded according to the criteria listed in the quality assurance plan checklist given below. The grade will be primarily based on the judgment of the instructor and TAs, but will include input both from the team that reviewed the QA plan and other members of the class as a whole.

2 Presentations (60 points)

You will present your quality assurance plan to the class during the final exam slot of finals week. Each team will likely have 10-15 minutes to present, followed by about 5 minutes of questions from the team assigned to review their work. The exact amount of time will be set immediately after the first draft of the QA plans are sent out.

As always with presentations, you should plan your time carefully to cover the most important parts of your quality assurance plan, leaving other details for the written document. Every student who did not present in mini project 1 must present in this forum.

Hints. To give you an idea of what we liked and disliked in QA plan presentations last year, here is a generic list of some of the positive comments and areas in need of improvement we identified on various plans.

Positives:

- Strong focus on spending time effectively, customizing techniques to resources
- Good description of review process
- Nice description of how process affects QA strategy
- Good consideration of time budget
- Driving plan via traceability matrix
- Gave schedule of how you were going to execute QA plan
- Good job setting context
- Specific quality goals
- Good consideration for threshold for completing testing
- Consideration of diverse issues (e.g. standards conformance)
- Good definition of scope—what is relevant to QA plan and what is not
- Detailed, comprehensive QA strategy
- Risk-based approach to choosing strategies
- Careful thought about QA tool usage and static analysis
- Good thought about focusing particular strategies on particular artifacts
- Good match of strategy to quality attribute

Negatives:

- Goals could be more specific and measurable
- Unrealistic to correct all defects
- Didn't give reasons for not using static analysis (OK not to use it, but explain why)
- Might consider how QA techniques should vary based on problem or artifacts
- Insufficient description of how testing was to be done
- Fairly heavyweight QA process, hard to tell if this will be feasible (please justify)
- Too much detail on process (e.g. individual requirements)
- Could be more discussion of quality attribute verification (vs. functional correctness)

3 Reviews (40 points)

The initial quality assurance plans are due May 3. The evening of May 3, the instructor will send your team the quality assurance plan and associated documentation of another team. Your team will carry out a formal technical review of this document. In reviewing the document, consider the size of team that produced it, using the 2-3 page per student rule of thumb listed above; documents produced by smaller teams are expected to be shorter and higher-level, and may omit some items such as precisely defined checklists.

You should use the checklist below, augmented (if you see fit) with other checklist items. All members of your team should do at least an hour of advance preparation reading the test plan and evaluating it with respect to the checklist. You may choose to have particular members of your team focus on particular elements of the plan, and/or particular parts of the checklist.

The formal technical review should last at least an hour, but no more than two hours. Carry out the process as described in lecture.

On the day of quality assurance plan presentations, you will turn in the following:

- For each team member, a list of issues identified. Each issue should identify the relevant place in the document, the issue, the severity (major or minor) and whether the issue is believed to be an error, an improvement, a clarification, a style issue, a question, or something else. Also, state the amount of time spent by that team member on the review.
- For the entire team, a review document. The document should include the name of the project being reviewed, the name of the review team, and the names of all the reviewers. State how much time was spent at the review meeting. The document should state who played the roles of moderator, reader, and recorder. Describe which, if any, checklist items you used in addition to those given below, and also any items you would recommend adding to the checklist for next year's class. The document should list all issues identified by the team, presented as described in the individual deliverable.

The review document should also include the team's overall judgment of the quality of the quality assurance plan. The scale is as follows:

- **Accept** - The quality assurance plan exceeds expectations, and in an industrial setting would be accepted with no follow up needed. All issues identified are minor.
- **Conditional Accept** - The quality assurance plan meets expectations overall, although there are some issues that must be addressed and verified. Most identified issues are minor, and if there are one or two major issues you believe the authors of the plan can address them without re-review. In an industrial setting, it would be accepted conditionally, with changes to be made and verified by a member of the inspection team (in this class, the instructor and TAs will do the verification).
- **Partial Reinspection** - The quality assurance plan is generally OK, but has some significant aspects that are missing or need significant revision. In addition to a number of minor issues, one or more major issues exist that are non-trivial to fix, and should be re-inspected before acceptance. In an industrial setting, there would likely be a reinspection of the parts of the document that were identified for major revision.

- **Rework and Reinspect** - The quality assurance plan is seriously flawed. The problems are significant enough that the entire document should be rewritten from the ground up and reinspected at a later date.

After each team presents, the team that reviewed their quality assurance plan will question the authoring team about the most important issues identified in the quality assurance plan.

Grading. The performance of the reviewing team will be graded according to how thorough their review of the document was; how well the document communicates the issues identified; how well reviewers were prepared (an individual criterion); how well the review team used the questioning time; and how well the overall judgment of the review team matched the judgment of the instructors, TAs, and class.

Hints. Last year, the best review questions given by students during the class were those that identified significant missing parts of a QA plan, or identified planned items that were not realistic or were poorly matched to the individual project. Unless there were major problems with the planned items, it is often more useful to think about what is missing from the plan than to criticize what is there. Finding nits in the plan is useful but less important than higher-level issues, especially for the limited question period.

4 Review Checklist

Format and Presentation.

- Are all elements of the required header and versioning information present?
- Is the document clearly written, easy to understand, and with correct usage of English?
- Is the project context given sufficient for you to evaluate the QA plan?
- Are necessary reference documents included, such as requirements, architecture, or design documents? Be understanding if they have not completed these documents yet, but ask yourself if they could have provided substitutes.

Goals and Strategy.

- Are the quality assurance goals clearly defined? Are they measurable? Are the goals quantitative where possible?
- Is the scope appropriate given the project?
- Are any major functional requirements or quality attribute requirements missing?
- Is the overview of strategy clear and appropriate to the project?

Approach.

- Is a quality assurance strategy specified for each of the goals given above?
- Is the strategy chosen for each goal appropriate for that functional requirement or quality attribute? Is the strategy explicitly and convincingly justified?
- Does the plan consider (not necessarily choose) aspects of testing, reviews, and static analysis? Does it combine multiple approaches in appropriate ways?

- Does the plan consider quality assurance at early phases of the software lifecycle?

Testing.

- Does the QA plan clearly define the testing approaches to be used?
- Does the QA plan define criteria for when testing is complete?
- Is the equivalence class analysis clear? Are there any equivalence classes or boundary cases missing? Does it consider both valid and invalid equivalence classes?
- Does the plan describe strategic considerations such as test scaffolding, integration testing, or regression testing?

Reviews.

- Does the plan define what will be reviewed, considering (but not necessarily choosing to review) both documents and code?
- Does the plan define the checklists to be used in the review? Are the checklists appropriate?
- Does the plan describe the kind of review and review process to be used?

Organization.

- Does the document describe the quality assurance process and how QA activities will be organized?
- Is the resource allocation reasonable with respect to both available resources and the quality assurance goals of the team? Are estimates of how much resources each task will take realistic?
- Is the management process adequate for ensuring that issues will be addressed? Are appropriate tools and processes chosen to coordinate various aspects of QA tasks?
- Are the deliverables and exit criteria clearly defined for each quality assurance task?

- Is the quality assurance process improvement plan, including applicable measurements, adequate?
- Are all processes and techniques proposed relevant and specific to the project?

5 Final Document (40 points)

Each team will receive comments from the reviewing team, and will revise their quality assurance plan to take the major identified issues into account. It is recommended that the revision section of the document describe changes that were made in response to each major point. You may disagree with the judgment of the review team; if so, explain why you disagree. If the review team simply misinterpreted an aspect of your document, you may not need to change the content but you should change the presentation to avoid misunderstandings if someone should read your document in the future.

You will be graded according to the checklist above, but special attention will be paid to how you have addressed the issues raised by the review team.