SE-4831: Software Quality Assurance

Lab 2: A Formal Inspection

Due: Defect Log due 23:00 December 11, 2013 End of Lab
Scribe Report: December 11, 2013 End of Lab

1. Introduction

Formal inspections represent one of the most effective ways of assuring software quality for delivered products. Depending upon the training of the inspectors and other factors, inspections have been shown to be extremely effective at identifying defects within software artifacts. In your past courses, namely SE280, you learned how to perform individual reviews. In this course, we have talked about formal inspections, and in lab today, we will be doing a formal Fagan inspection on an artifact of the Digital Home System.

In this particular inspection, the author is not able to be present. Thus, you will be focusing on complete the inspection without the author being present. The artifact that is to be inspected is the Digital Home System SRS version 1.2, available in the archived location http://www.softwarecasestudy.org/pdfdoc/DH-SRS-ver1.2-01072011.pdf.

2. Lab Process

The lab will begin by breaking into your lab teams. The instructor has assigned roles to each team member. The moderator of the activity is responsible for briefing the inspection team on the scope of their assignments and the purpose for the inspection. This meeting should take approximately 10 minutes or less.

After the overview meeting is completed, each individual will spend 35 minutes doing individual inspections on the assigned segments of the SRS artifact. There is a checklist available on the course website that should be used for this purpose. When going through the artifact, each individual (moderator and scribe included) should complete their own inspection defect log using the pdf form linked to from the course website.

After individual preparation has been completed, inspections teams should reassemble to go through the artifact. The moderator is responsible for leading the review, going paragraph by paragraph through the artifact and highlighting the defects that are found.

While this is going on, the scribe should be recording the defects found in the Excel recording spreadsheet. The spreadsheet has rows for each defect and columns for the defect type, the defect location, the defect description, and the inspectors who found the defect. It is very important that the results show who found each defect. This meeting should last approximately 35 minutes as well.

When this meeting is completed, the scribe should be prepared to present to the class all the findings from their inspection. No materials need to be prepared. Rather the presentation will simply involve going through the defect log, discussing the defects as a class, and combining the

results into a final Google document showing all findings from all inspection teams as well as information about which team found the mistakes.

3. Deliverables

3.1. Individual

Each individual should submit in hard copy their personal inspection log. This is due by the close of the lab session.

3.2. As a team

The team is responsible for submitting the defect log from the inspection meeting. This most logically should be assigned to the scribe. This is due by the close of the lab session.

4. Inspection Teams

	Moderator	Scribe	Inspector 1	Inspector 2	Inspector 3	Inspector 4
Team 1	Johnsen, Bradley James	Breckenridge, Keith Douglas	Desai, Mrudang	Nemetz, Keenan Phillip	Raiche, Duane	
Team 2	Singh, Barinderpal	Cook, Andrew Arden	Everts, Dosty	Avery, Joshua		
Team 3	Magyar, Tyler Michael	Campbell, Jonathan Lynn	Ramirez, Brandon	Graef, Ryan		
Team 4	Berger, Andrew	Grodek, David	Curtis Williams, Michael	Stewart, Emil Keith	Sanfelipo, Isabella Louise	
Team 5	Delfanian, Reekah Adrian	Parrish, Benjamin Adam	Johnson, Austin David	Dixon, Matthew	Billetdeaux, Greg J	Reiland, Andrew Devin
Team 6	Reyes, Joseph Evan	Hackbarth, Tyler James	Dorus, Michael Stephen	Kaas, Peter	Wasielewski, Thaddeus H	
Team 7	Askow, Nathan	Flanery, James William Francis	Gieske, Nikolas C	Franklin, Justin Tyler	Altschwager, Michael Matthew	
Team 8	AlMatroudi, Suliman	Heinz, Eric Randall	Kostecki, Eric J	Harris, Frank Paul	Nguyen, Duc Hoai	

Odd Teams SRS Sections 1-3
Even Teams SRS Sections 4-5

DigitalHome Inspection Process	
Purpose	Guide for carrying out a formal Fagan-type inspection of a software work product.
Entry Criteria	Software artifact appropriate for inspection
Roles	• Author: The person responsible for development of the softer work product • Moderator: The person responsible for managing the inspection process and facilitating the inspection meeting. • Recorder: The person responsible for recording inspection meeting decisions and completing the inspection report. • Inspector: the person who inspects the work product or some portion of it.
Phase	**Activity**
Planning	The inspection is planned by the moderator
Overview Meeting	• The author describes the background of the work product and provides an overview of its contents. • The moderator reviews the inspection guidelines (listed below), describes the inspection process and inspection forms, assigns roles and makes inspection assignments.
Preparation	Each inspector uses the inspection checklist to examine the work product to identify possible defects and record them in a defect log.
Inspection Meeting	During this meeting • The moderator reads through the work product, part by part • The inspectors point out possible defects found in each part. • The inspection team decides on the defect status (no defect, major defect, minor defect) • The recorder records information about defects in a defect log and completes inspection report.
Rework	The author makes changes to the work product based on the defects indentified in the inspection meeting.
Follow-Up	The changes by the author are checked to make sure all defects have been addressed.
Exit Criteria	□ All inspection forms are complete. □ Software work product has been revised, removing identified defects.
Inspection Guidelines	□ Plan and prepare for the inspection. □ Develop an inspection checklist (or revise an existing one) of things to be checked. □ The work product should be inspected in small “chunks”. □ Inspect the work product, not the developer. □ Set an agenda for inspection meeting and stick to it. □ The inspection meeting should last no more than one hour. □ Limit the number of participants and insist upon advanced preparation. □ Limit debate and rebuttal in the inspection meeting. □ Identify problems; do not attempt to solve them. □ Take written notes of the meeting; collect size, effort and defect data. □ Complete all inspection forms.

.

SRS Checklist and Requirements Defect Types

Type	Organization and Completeness
O1	Are all internal cross-references to other requirements correct?
O2	Are all requirements written at a consistent and appropriate level of detail?
O3	Do the requirements provide an adequate basis for design?
O4	Is the implementation priority of each requirement included?
O5	Are all external hardware, software, and communication interfaces defined?
O6	Have algorithms intrinsic to the functional requirements been defined?
O7	Does the specification include all of the known customer or system needs?
O8	Is the expected behavior documented for all anticipated error conditions?
O9	Are there problems in the organization of the SRS? Are the supporting non-requirement parts of the SRS (e.g., purpose/scope sections, reference sections, etc.) correct and complete?
O10	Are there requirements included which were not requested by the customer?
O11	Is a requirement statement compound – does it contain more than one identifiable requirement?
	Correctness
C1	Do any requirements conflict with or duplicate other requirements?
C2	Is each requirement written in clear, concise, unambiguous language?
C3	Is each requirement verifiable by testing, demonstration, review, or analysis?
C4	Is each requirement in scope for the project?
C5	Is each requirement free from content and grammatical errors?
C6	Is any necessary information missing from a requirement? If so, is it identified as TBD?
C7	Can all of the requirements be implemented within known constraints?
C8	Are any specified error messages unique and meaningful?
	Quality Attributes
Q1	Are all performance objectives properly specified?
Q2	Are all security and safety considerations properly specified?
Q3	Are other pertinent quality attribute goals explicitly documented and quantified, with the acceptable tradeoffs specified?
	Traceability
T1	Is each requirement uniquely and correctly identified?
T2	Is each software functional requirement traceable to a higher-level requirement (e.g., system requirement, use case)?
	Special Issues
S1	Are all requirements actually requirements, not design or implementation solutions?
S2	Are all time-critical functions identified, and timing criteria specified for them?
S3	Have internationalization issues been adequately addressed?
S4	Are there sections related to IEEE 830-1998 that should be included?

Note: This list is based on material from Peer Reviews in Software by Karl E. Wiegers (Addison-Wesley, 2001)