Enterprise Business Requirements 32569

Autumn 2018

Tutorial 6 and 7 – Requirements Validation and Inspection

Tutorial Objectives:

- To consolidate your understanding of requirements validation and inspection.
- To practice with requirements validation and inspection exercises.

In this tutorial exercise you will work with your group to conduct a semi-formal inspection of the SRS and use cases for the Ski Resort Information System case study. An example of inspection checklist for use cases and requirements specifications is included in this file (below) for your information. Your review process should be structured by the items in the checklists.

For each of the problems that you will identify, you must provide an appropriate recommendation on how to fix it. You can use a table similar to the one included in this file (below) to document your findings when checking the requirements or use cases against the quality attributes that was covered in the lecture. If you have problems understanding the items in the checklist or how to provide a recommendation, ask your tutor.

1. You must familiarise yourself with the appropriate documents that are going to be inspected. These documents are provided in the tutorial folder on UTS online. (10 min)

2. Work with your group using the appropriate checklist to structure your discussion. Find issues and defects in the documents. Write them down. Your tutor will visit each group and answers any questions you may have. (20 min)

3. Your tutor will call each group in the class to present 2 to 3 defects/errors they have found. Each group should try to present unique issues and defects that other teams have not mentioned. (15 Min)

Note: It is possible that not all of the items in the attached checklists can be checked against for this particular case study. You must identify those items and explain why they cannot be checked.

Inspection Checklist for Use Case Documents

- Is the use case a standalone, discrete task?
- Is the goal, or measurable value, of the use case clear?
- Is it clear which actor(s) benefit from the use case?
- Is the use case written at the essential level, rather than as a specific scenario?
- Is the use case free of design and implementation details?
- Are all anticipated alternative courses documented?
- Are all known exception conditions documented?
- Are there any common action sequences that could be split into separate use cases?
- Is the dialog sequence for each course clearly written, unambiguous, and complete?
- Is every actor and step in the use case pertinent to performing that task?
- Is each course defined in the use case feasible?
- Is each course defined in the use case verifiable?
- Do the pre- and post-conditions properly frame the use case

Checklist for Requirements Specification Reviews

Organization and Completeness

- ❑ Are all internal cross-references to other requirements correct?
- ❑ Are all requirements written at a consistent and appropriate level of detail?
- ❑ Do the requirements provide an adequate basis for design?
- ❑ Is the implementation priority of each requirement included?
- ❑ Are all external hardware, software, and communication interfaces defined?
- ❑ Have algorithms intrinsic to the functional requirements been defined?
- ❑ Does the specification include all of the known customer or system needs?
- ❑ Is the expected behavior documented for all anticipated error conditions?

Correctness

- ❑ Do any requirements conflict with or duplicate other requirements?
- ❑ Is each requirement written in clear, concise, unambiguous language?
- ❑ Is each requirement verifiable by testing, demonstration, review, or analysis?
- ❑ Is each requirement in scope for the project?
- ❑ Is each requirement free from content and grammatical errors?
- ❑ Is any necessary information missing from a requirement? If so, is it identified as TBD?
- ❑ Can all of the requirements be implemented within known constraints?
- ❑ Are any specified error messages unique and meaningful?

Quality Attributes

- ❑ Are all performance objectives properly specified?
- ❑ Are all security and safety considerations properly specified?
- ❑ Are other pertinent quality attribute goals explicitly documented and quantified, with the acceptable tradeoffs specified?

Traceability

- ❑ Is each requirement uniquely and correctly identified?
- ❑ Is each software functional requirement traceable to a higher-level requirement (e.g., system requirement, use case)?

Special Issues

- ❑ Are all requirements actually requirements, not design or implementation solutions?
- ❑ Are all time-critical functions identified, and timing criteria specified for them?
- ❑ Have internationalization issues been adequately addressed?