Requirements and Specification, ESOF 328, Spring 2020
“Requirements management practices” (Chapter 27)
“Change happens” (Chapter 28)
“Software requirements and risk management” (Chapter 32)
Jan. 15

Requirements management practices, Chapter 27

Activities which should be defined and documented: (page 458)

- Tool, techniques and conventions for distinguishing versions of individual requirements
- How requirements will be approved and baselined
- How new requirements and changes to existing requirements are proposed, evaluated, negotiated, and communicated
- How to assess the impact of a proposed change
- Requirements attributes and requirements status-tracking procedures, including the requirements statuses that you will use and who can change them
- Who is responsible for updating requirements trace information and when
- How to track and resolve requirements issues
- How the project’s plans and commitments will reflect requirements changes
- How to use the requirements management (RM) tool effectively

Requirements baseline is a set of requirements that stakeholders have agree to. Subsequent changes can be made only through the project’s defined change control procedure.

Requirement attributes: (page 462)

- Date the requirements was created
- Current version number of the requirements
- Author who wrote the requirement
- Priority
- Status
- Origin or source of the requirement
- Rationale behind the requirement
- Release number or iteration to which the requirement is allocated
- Stakeholder to contact with questions or to make decisions about proposed changes
- Validation method to be used or acceptance criteria

Possible requirement statuses: (Page 465)

- Proposed
- In progress
- Drafted
- Approved
- Implemented
- Verified
- Deferred
- Deleted
- Rejected

Common Types of Requirements Issues: (Table 27-2, page 467)

- Requirement question
- Missing requirement
- Incorrect requirement
- Implementation question
- Duplicate requirement
- Unneeded requirement

Recall from before, common requirements problems:

Omitting requirements – missing requirement from above
Ambiguous requirements – requirement question, duplicate requirements,
Missing stakeholders
Including design
Gold platting
Scope creep (more of a problem after baselining)

Change happens, Chapter 28

Use change management process that ensures that: (page 471-472)

- Proposed requirements changes are thoughtfully evaluated before being committed to
- Appropriate individuals make informed business decisions about requirement changes
- Change activity is made visible to affected stakeholders
- Approved changes are communicated to all affected participants
- The project incorporates requirements changes in a consistent and effective fashion

Scope creep – project continuously incorporates more functionality without adjusting resources, schedules, or quality goals.

To manage scope creep document: (page 473)

- Business objectives
- Product vision
- Project scope
- Limitations of the new system

Change control policy: (page 474)

- All changes must follow the process. If a change request is not submitted in accordance with this process, it will not be considered.
- No design or implementation work other than feasibility exploration will be performed on unapproved changes
- Simply requesting a change does not guarantee that it will be made. The project's change control board (CCB) will decide which changes to implement.
- The content of the change database must be visible to all project stakeholders.
- Impact analysis must be performed for every change.
- Every incorporated change must be traceable to an approved change require.
- The rationale behind every approval or rejection of a change request must be recorded.

Change Control Board (CCB) is the body of people that decides which proposed changes and new requirements to accept, which to accept with revisions, and which to reject.

Agile projects have two choices about how to handle changes within an iteration: (page 489)

- Freeze the baseline for an iteration once it is under way
- Introduce high-priority changes as soon as you learn about them

Goal is to avoid both excessive change (churning requirements) and excessive rigidity (frozen requirements) within an iteration

Agile team is a collaborative and cross-functional group of developers, testers, a business analyst, a project manager, and others. The team is already configured like the change control board.

Software requirements and risk management, Chapter 28

Risk – condition that could cause some loss or otherwise threaten the success of a project

If something has occurred, that isn't a risk, it is an issue and should be dealt with using the requirements management plan.

Elements to risk management:

- Assessment (identification, analysis and prioritization)
- Avoidance
- Control (management planning which includes mitigation approaches, contingency plans, owners and timelines, resolution which includes executing the mitigation plan, monitoring)

Express risks using the format: condition – consequence

Risk exposure – probability that the risk occurs (somewhere between 0, impossible, to 1, inevitable) * impact if risk occurs (from 1, no problem, to 10, big problem)

Don't assume that risks are under control just because they are identified and have mitigation actions. Must monitor the risk status and determine if the mitigation plans work (page 542)