# Software Requirements Management

Software Requirements, Second Edition by Karl E. Wiegers

# Requirements Management Principles and Practices

- Controlling changes to the requirements baseline
- Keeping project plans current with the requirements
- Controlling versions of both individual requirements and requirements documents
- Tracking the status of the requirements in the baseline
- Managing the logical links between individual requirements and other project work products

# Requirement Attributes

- Date the requirement was created
- Its current version number
- Author who wrote the requirement
- Person who is responsible for ensuring that the requirement is satisfied
- Owner of the requirement or a list of stakeholders (to make decisions about proposed changes)
- Requirement status
- Origin or source of the requirement

# Requirement Attributes

- The rationale behind the requirement
- Subsystem (or subsystems) to which the requirement is allocated
- Product release number to which the requirement is allocated
- Verification method to be used or acceptance test criteria
- Implementation priority
- Stability (an indicator of how likely it is that the requirement will change in the future; unstable requirements might reflect ill-defined or volatile business processes or business rules)

# Tracking Requirements Status

- Proposed
- The requirement has been requested by an authorized source.
- Approved
- The requirement has been analyzed, its impact on the project has been estimated, and it has been allocated to the baseline for a specific release. The key stakeholders have agreed to incorporate the requirement, and the software development group has committed to implement it.
- Implemented
- The code that implements the requirement has been designed, written, and unit tested. The requirement has been traced to the pertinent design and code elements.

- Verified
- The correct functioning of the implemented requirement has been confirmed in the integrated product. The requirement has been traced to pertinent test cases. The requirement is now considered complete.
- Deleted
- An approved requirement has been removed from the baseline. Include an explanation of why and by whom the decision was made to delete it.
- Rejected
- The requirement was proposed but is not planned for implementation in any upcoming release. Include an explanation of why and by whom the decision was made to reject it.

# Measuring Requirements Management Effort

- Count the effort devoted to the following activities as requirements management effort:
- Submitting requirements changes and proposing new requirements
- Evaluating proposed changes, including performing impact analysis
- Change control board activities
- Updating the requirements documents or database
- Communicating requirements changes to affected groups and individuals
- Tracking and reporting requirements status
- Collecting requirements traceability information

# Change Happens

# Managing Scope Creep

# Change-Control Policy

- All requirements changes shall follow the process. If a change request is not submitted in accordance with this process, it won't be considered.
- No design or implementation work other than feasibility exploration shall be performed on unapproved changes.
- Simply requesting a change doesn't guarantee that it will be made. The project's change control board (CCB) will decide which changes to implement. (We'll discuss change control boards later in this chapter.)
- The contents of the change database shall be visible to all project stakeholders.
- The original text of a change request shall not be modified or deleted.
- Impact analysis shall be performed for every change.
- Every incorporated requirement change shall be traceable to an approved change request.
- The rationale behind every approval or rejection of a change request shall be recorded.

# The Change Control Board

- The CCB membership should represent all groups who need to participate in making decisions within the scope of that CCB's authority. Consider selecting representatives from the following areas:
- Project or program management
- Product management or requirements analyst
- Development
- Testing or quality assurance
- Marketing or customer representatives
- User documentation
- Technical support or help desk
- Configuration management

# Measuring Change Activity

- The number of change requests received, currently open, and closed
- The cumulative number of changes made, including added, deleted, and modified requirements (You can also express this as a percentage of the total number of requirements in the baseline.)
- The number of change requests that originated from each source
- The number of changes proposed and made in each requirement since it was baselined
- The total effort devoted to processing and implementing change requests
- The number of cycles through the change process that it took to correctly implement each approved change (Sometimes changes are implemented improperly or cause other errors that need to be corrected.)

# Change Isn't Free: Impact Analysis

- **Impact Analysis Procedure**
- The CCB Chair will typically ask a knowledgeable developer to perform the impact analysis for a specific change proposal. Impact analysis has three aspects:
- Understand the possible implications of making the change. Change often produces a large ripple effect. Stuffing too much functionality into a product can reduce its performance to unacceptable levels, as when a system that runs daily requires more than 24 hours to complete a single execution.
- Identify all the files, models, and documents that might have to be modified if the team incorporates the requested change.
- Identify the tasks required to implement the change, and estimate the effort needed to complete those tasks.
- Trap  Skipping impact analysis doesn't change the size of the task. It just turns the size into a surprise. Software surprises are rarely good news.