Software Requirements: 7 Critical Success Factors

Sponsored by:

Karl Wiegers
Principal Consultant, Process Impact
www.processimpact.com
iRise is an enterprise visualization software and services company that helps business and IT departments better communicate, collaborate, and ultimately deliver better software in less time. iRise is the only solution that allows all stakeholders to collaborate, adapt and innovate on their vision in real time throughout the entire software development process.
Featured Speaker

Karl Wiegers
Principal Consultant, Process Impact
www.processimpact.com

Process Impact

Phone #: 503-698-9620
E-mail: karl@processimpact.com
Coming August 22, 2013

- **Software Requirements, 3rd Ed.** by Karl Wiegers and Joy Beatty
- Many enhancements over 2E:
  - much more on elicitation, quality attributes, business requirements, role of the BA, writing excellent requirements
  - new chapters on data requirements, requirements reuse, agile projects
  - many sections on applying requirements practices on agile projects
  - chapters on requirements for project classes: enhancement & replacement, outsourced, packaged solutions, embedded systems, business process automation, business analytics
**CSF #1: A shared understanding of what requirements are and why we care**

- People use many terms for various types of requirements
- Leads to confusion and headaches
  - What kind of information is it?
  - How do we get it?
  - Who is responsible for it?
  - Where does it go?
  - What do we do with it?
- Not everyone believes requirements are important (or necessary!)
What Is a “Software Requirement”?

“Requirements are ... a specification of what should be implemented. They are descriptions of how the system should behave, or of a system property or attribute. They may be a constraint on the development process of the system.”

-- Ian Sommerville & Pete Sawyer
Three Levels of Software Requirements

- Business Requirements
- User Requirements
- Functional Requirements
- Business Rules
- Quality Attributes
- External Interfaces
- Constraints

Vision and Scope Document
User Requirements Document
Software Requirements Specification
Good Requirements Accelerate Development

- Business requirements establish a shared vision, goals, and expectations.
- User involvement reduces chance of system rejection.
- Requirements ensure that the functionality built enables users to perform essential business tasks.
- Requirements define achievable quality expectations.
- Emphasizing requirements is cheaper than beta testing.
- The right changes are incorporated appropriately.
- Reviews detect and remove errors early.
CSF #2: Clearly defined business requirements

◆ Business Objectives
  ● come from business problems or market opportunities
  ● define success metrics for customer value

◆ Product Vision
  ● concept of what the product might eventually become
  ● identifies business benefits the system will provide

◆ Project Scope
  ● boundary between what’s in and out
  ● include limitations and exclusions
  ● facilitates making project commitments and prioritization
  ● prerequisite to managing scope creep
Four Techniques for Scope Definition

◆ **Context Diagram**
  - shows external entities
  - no internal system details

◆ **Use Case Diagram**
  - shows use case–actor connections

◆ **Feature Roadmap**
  - describe levels of feature enrichment
  - plan specific feature levels for each release

◆ **System Events**
  - triggers that stimulate some system response
  - could be business events, temporal events, signal events
CSF #3: Trained, skilled, and motivated business analysts

- Sponsor
- Business requirements
- Size and complexity information
- Project manager
- User reps
- User requirements
- Development
- Functional and nonfunctional requirements
- Other stakeholders
- Expectations and constraints
- Functional and nonfunctional requirements
- Testing
- Approved
The Business Analyst - 1

◆ Plays an essential communication role
  ● talk to users: application domain
  ● talk to developers: technical domain
  ● translate user requirements into technical requirements

◆ Performs many project tasks
  ● define business requirements
  ● identify project stakeholders and user classes
  ● elicit requirements
  ● analyze requirements
  ● document requirements
  ● lead requirements validation
  ● facilitate requirements prioritization
  ● manage requirements
The Business Analyst - 2

- Needs many skills
  - listening
  - interviewing and questioning
  - analytical
  - communication and interpersonal
  - systems thinking
  - facilitation
  - leadership
  - observation
  - writing
  - modeling
  - organization
  - creativity
CSF #4: A collaborative partnership with customers

- Stakeholders
  - Customers
  - Other Stakeholders
    - Direct and Indirect Users
      - Favored User Classes
      - Disfavored User Classes
      - Ignored User Classes
      - Other User Classes
    - Other Customers
The Need for Customer Involvement

Customer involvement is the most critical factor in software quality.

Expectation Gap
(Surprise!)
Building the Collaborative Partnership

- Identify stakeholders.
- Identify user classes.
- Select “product champions” to represent each user class.
- Co-locate BA, developers, and customer reps if possible.
- Agree on how you will work together.
- Agree on customer rights and responsibilities regarding requirements.
The Product Champion

- Primary interface between development and customer communities
- Ideally, a real user; not a user manager, funding sponsor, or simulated user
- Reconciles incompatible user requirements
- Goal is to present developers with a unified set of requirements
- Must be empowered to make binding decisions
- Document product champion responsibilities
CSF #5: Rigorous and ongoing requirements prioritization

- Not everything can be top priority!
- Setting priorities will help you:
  - work on the right things first
  - make tradeoff decisions
  - deal with added and changed requirements
- Need to bypass politics and emotion
  - favorable indicator: customer value (benefit + penalty)
  - unfavorable indicators: cost and technical risk
- Need to understand which requirements are most important and most urgent.
A Prioritization Scale

- **Urgent**
  - **Important**
    - High Priority: must be included in next release
  - **Not Important**
    - Low Priority: would be nice to have if we can fit it in

- **Not Urgent**
  - **Important**
    - Medium Priority: must be included, but can wait for later release
  - **Not Important**

[Covey, Stephen R. The 7 Habits of Highly Effective People. Free Press, 2004.]
**CSF #6: Taking an incremental and iterative approach to requirements development**

- You can’t get all requirements right up front
- Start at high level of abstraction and work down:
  
  *business reqs*  =>  *user reqs*  =>  *functional reqs*

- Do first-cut elicitation, prioritize, then flesh out top-priority requirements for the next release or iteration
- Expect to revisit requirements to refine, clarify, extend, and modify
- Key concept: *progressive refinement of details*
Components of Requirements Engineering

- **Requirements Engineering**
  - **Requirements Development**
  - **Requirements Management**

**Elicitation** → **Analysis** → **Specification** → **Validation**

- Clarify
- Close gaps
- Re-evaluate
- Rewrite
- Confirm and correct
CSF #7: Anticipating and accommodating change

◆ Anticipate Change
  ● Requirements grow – deal with it.
  ● Know the users and their business drivers.
  ● Identify likely sources of change.

◆ Accommodate Change
  ● Understand your team’s capacity to adapt to change.
  ● Structure applications to facilitate rapid change.

◆ Manage Change
  ● Use a tool to manage change requests.
  ● Follow your change control process.
  ● Evaluate impact before approving a change request.
Some Change Management Strategies

◆ **Requirements Development**
  - adopt best practices for elicitation and customer engagement
  - define project scope
  - define requirements baselines

◆ **Change Control System**
  - define a change control process
  - establish a change control board for each project
  - align changes with business objectives

◆ **Project Management**
  - follow appropriate development life cycles
  - incorporate contingency buffers into schedules
  - perform formal risk management
Your Next Steps

◆ Assess points of pain.
◆ Understand root causes of problems.
◆ Define effective practices to address the pain.
  ● industry “good practices”
  ● local positive experiences
◆ Develop procedures, templates, work aids
◆ Train project participants.
◆ Share your successes, learn from each other.
◆ Communicate, communicate, communicate!
Software Requirements: 7 CSFs

NO

SURPRISES!
White Paper Download: 
*Building a Business Case for Visualization*

Register for our next webinar: June 27th, 10:00 AM PDT Requirements Visualization and the SDLC: “A Day in the Life”
https://gotomeeting.rsvp1.com/s15d09bJrwnp

- Help define project scope
- Increase communication and collaboration among team
- Get a more precise estimation of resources, timeline, and overall project cost
- Show concept for executive approval
- Validate proof of concept with business leaders
- Get a 30-day Free iRise Design Studio Trial at www.irise.com
Q & A

**Speaker**
Karl Wiegers
Process Impact
karl@processimpact.com
www.processimpact.com

**Sponsor**
Franklyn Carr
iRise
fcarr@irise.com
www.iRise.com