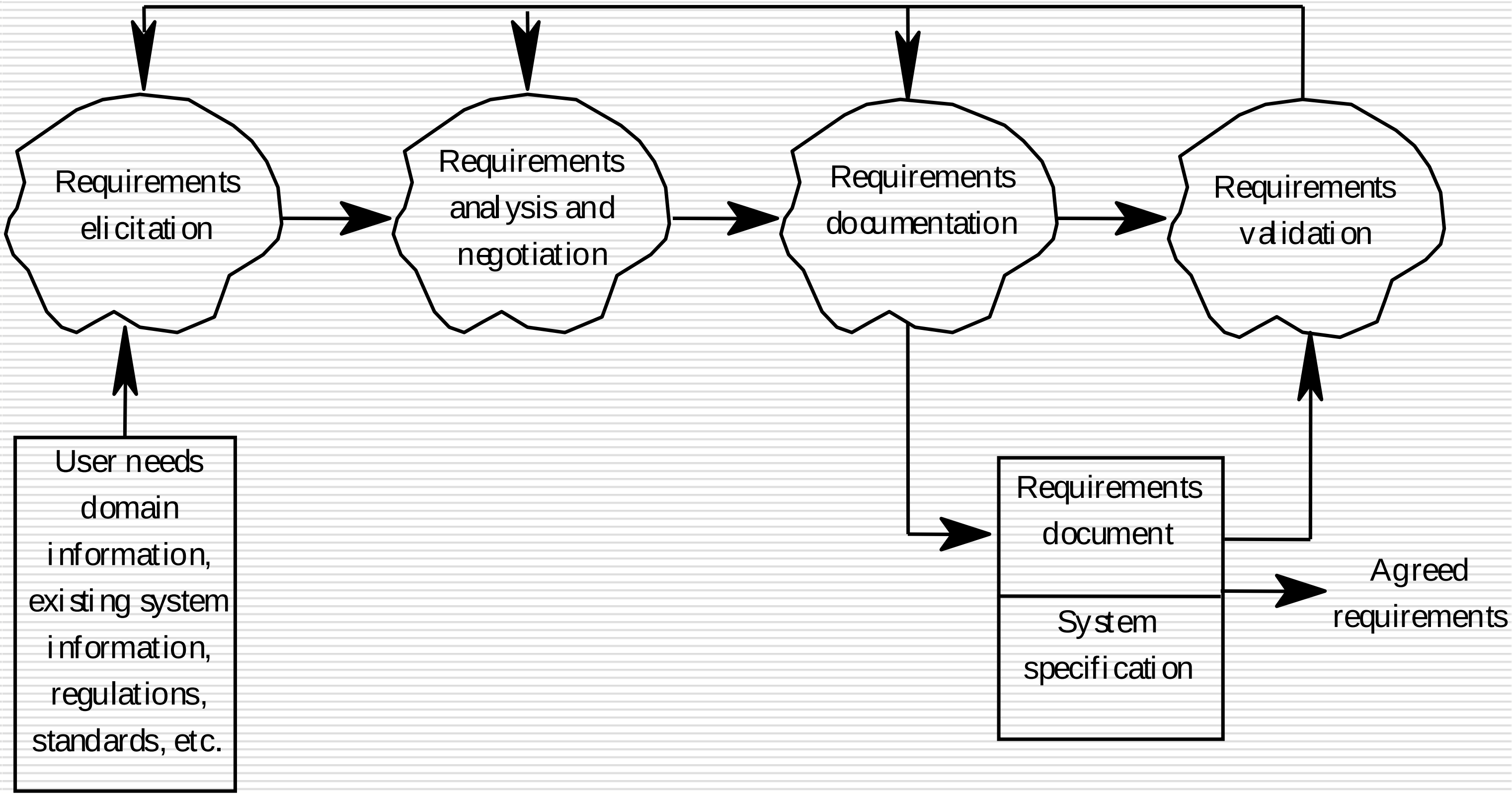
Requirements Process

Lam Quang Vu – SE Dept. HCMUS

Sample Process (Req Development):


Requirements elicitation
Requirements analysis and negotiation
Requirements documentation
Requirements validation
User needs domain information, existing system information, regulations, standards, etc.
Requirements document
System specification
Agreed requirements

Elicitation⬄Analysis⬄Negotiation:

- Characteristics:
  - Iterative
    - Step by Step, not one big step.
  - Inter-dependent processes
    - Each step depends on each other
  - Repeated
    - Each step has to be repeated
- Sample Outputs
  - Final Proposal
    - Agreed Requirements
  - Vision Document
  - Software Requirements Specification
  - High Level Design Models
    - UML
    - Network Diagrams
    - Object Models

Elicitation - Techniques

- **Brainstorming**: Brainstorming sessions are used to let the stakeholders come up with creative ideas or new approaches to a problem
- **Workshops:** Workshops are facilitated meetings with multiple stakeholders to draw out and document requirements.

Elicitation - Techniques

- **Interviewing**: Interviews are in-person meetings where the business analyst asks questions to get information from the stakeholder.
- **Surveys**: Surveys are used to gather information anonymously from the stakeholders.

Elicitation - Techniques

- **Documentation Review**: This is the process of obtaining requirements from written documentation such as manuals.
- **Prototyping**: This is the use of partially finished versions of the software that have been created to help validate requirements

Elicitation - Techniques

- **Focus Groups**: Focus Groups are group interviews where the business analyst raises issues and questions to obtain information from the stakeholders.
- **Observation**: Observation is when the business analyst watches the users performing their daily tasks and asks questions about the tasks and work.

Elicitation - Techniques

- **Card sorting** - useful if you want to understand the user's classification of his/her knowledge domain;
- **Role playing**: Carrying out a user role yourself

Technique: Interviewing

- Simple direct technique
- Context-free questions can help achieve bias-free interviews
- Then, it may be appropriate to search for undiscovered requirements by exploring solutions.
- Convergence on some common needs will initiate a “requirements repository” for use during the project.
- A questionnaire is no substitute for an interview.

Interview: Context free question

- Goal is to prevent prejudicing the user’s response to the questions.
- Examples:
  - Who is the user?
  - Who is the customer?
  - Are their needs different?
  - Where else can a solution to this problem be found?
- Context-free questions also parallel the questions salespeople are taught to ask as part of a technique called “solutions selling.”
- After context-free questions are asked, suggested solutions can be explored.

Interview: Show time

- Establish Customer or User Profile
- Assessing the Problem
- Understanding the User Environment
- Recap the Understanding
- Analyst’s Inputs on Customer’s Problems
- Assessing Your Solution (if applicable)

Technique: Requirements Workshop

- The requirements workshop is perhaps the most powerful technique for eliciting requirements.
- It gathers all ***key*** stakeholders together for a short but intensely focused period.
- The use of an outside facilitator experienced in requirements management can ensure the success of the workshop.
- Brainstorming is the most important part of the workshop.

Preparing for the workshop

- Selling the workshop *concept* to stakeholders
- Ensuring the Participation of the Right Stakeholders
- Logistics
  - Whiteboard, projector
  - Drinks / Snacks
- Warm-up materials
  - Project-specific information
  - Out-of-box thinking preparation

Role of the Facilitator

- Establish professional and objective tone to the meeting.
- Start and stop the meeting on time.
- Establish and enforce the “rules” for the meeting.
- Introduce the goals and agenda for the meeting.
- Manage the meeting and keep the team “on track.”
- Facilitate a process of decision and consensus making, but avoid participating in the content.
- Make certain that all stakeholders participate and have their input heard.
- Control disruptive or unproductive behavior.

Workshop Agenda

- Set an agenda before the workshop and publish it along with the other pre-workshop documentation.
- Balance is the key, try to stay on the agenda, but do not strictly obey it, especially if good discussion is going on.
- Order lunch in, and have a *light* working lunch. :-)

Running the Workshop

- Allow for human behavior, and have fun with it.
  - Do not “attack” other members.
  - Do not get on a soap box.
  - Do not come back late from a break.
- Workshop Rules
  - Late after break
  - Talk too much
  - No talk at all

Workshop Problems and Suggestions

- Time Management
  - It's difficult to get going after breaks and lunch.
  - Key shareholders may be late returning
- Grandstanding, domineering positions
- Lack of input from stakeholders
- Negative comments, petty behaviors, and turf wars
- Flagging energy after lunch

- Facilitator keeps a timer for all breaks and fines anyone that is late, everyone gets one free pass.
- Everyone gets one 5 minute position statement.
- Facilitator encourages everyone to use 5-minute position and great idea ticket.
- Lite lunches, afternoon breaks, rearrange seating

Requirements Elicitation Guidelines[1]

- □ Assess System Feasibility
- □ Be sensitive to organizational and political considerations
- □ Identify and consult system stakeholders
- □ Record requirements sources
- □ Use Business concerns to drive requirements elicitation
- □ Look for domain constraints
- □ Record requirements rationale
- □ Collect requirements from multiple viewpoints
- □ Prototype poorly understood requirements
- □ Use scenarios to elicit requirements
- □ Define operational processes
- □ Reuse requirements

Identify and Consult System Stakeholders

- If lacking consideration of everyone who is likely to be affected by the introduction of the system, there is a great likelihood of missing some critical requirements.

- “Identifying stakeholders and discussing the system with them makes people feel like they are part of the requirements elicitation process. In fact, it *makes* them a part of it.”

Use Business Concerns to Drive Requirements Elicitation

- If a system is to be useful, it must contribute to the key concerns of the business. If the concerns are identified and used as drivers of the requirements elicitation process, there will be higher confidence that the system will meet real organization needs.

- Making the business concerns explicit helps to focus and clarify these goals.

Collect Requirements from Multiple Viewpoints

- If requirements are collected from a single viewpoint, they are unlikely to meet other stakeholders' requirements.
- Collecting requirements from multiple viewpoints is a useful way to prioritize requirements
- Identified viewpoints can be used to help
  - organize requirements elicitation and
  - organize the requirements specification, too.

Reuse Requirements

- Saves money and time. Studies have shown that similar systems can re-use up to 80% of the requirements.
- Reuse reduces risk. Reused requirements have a better chance of being understood by all the stakeholders.
- Requirements reuse may lead to additional reuse in other lifecycle activities.
  - Component design
  - Tests
  - Code

Technique: Brainstorming

- Brainstorming involves both idea *generation* and idea *reduction*.
- The most creative, innovative ideas often result from combining, seemingly unrelated ideas.
- Various voting techniques may be used to prioritize the ideas created.
- Although live brainstorming is preferred, web-based brainstorming may be a viable alternative in some situations

Rules for Brainstorming

- Do not allow criticism or debate.
- Let your imagination soar
- Generate as many ideas as possible
- Mutate and combine ideas
- Idea Reduction
  - Pruning ideas that are not worthy of further discussion
  - Grouping of similar ideas into one super topic
- Prioritize the remaining ideas

Technique: Storyboarding

- The purpose of storyboarding is to elicit early “Yes, But” reactions.
- Storyboards can be positive, active, or inactive.
- Storyboards identify the players, explain what happens to them, and describes how it happens.
- Make the storyboard sketchy, easy to modify, and unshippable.
- Storyboard early and often on every project with new or innovative content.

Technique: Use Cases

- Use Cases, like storyboards, identify the who, what, and how of system behavior.
- Use Cases describe the interactions between a user and a system, focusing on what they system “does” for the user.
- The Use Case model describes the totality of the systems *functional* behavior.
- Early stages: After you have an overview of the use cases, perhaps only by a phrase apiece, expand 10% of them in detail.
- More later …

Technique: Role Playing – variant on use cases

- Role playing allows stakeholders to experience the user’s world from the user’s perspective.
- A scripted walkthrough may replace role playing in some situations, with the script becoming a live storyboard.

(Class-Responsibility-Collaboration (CRC) cards, often used in object-oriented analysis, are a derivative of role playing.)

Technique: Prototyping

- Prototyping is especially effective in addressing the “Yes, But” and the “Undiscovered Ruins” syndromes.
- A software requirements prototype is a partial implementation of a software system, built to help developers, users, and customers better understand system requirements.
- Prototype the “fuzzy” requirements: those that, although known or implied, are poorly defined and poorly understood.

Technique: Survey

- Refs:
  - http://www.surveysystem.com/sdesign.htm
  - http://www.surveyworld.org/good_survey.php?t=4
- Tools:
  - Google spread
  - http://www.createsurvey.com/
  - http://www.makesurvey.net/

Refs

- □ http://en.wikipedia.org/wiki/Software_prototyping