Hybrid Method for Formalizing WinWin Stakeholder Inputs

Hasan Kitapci
USC Center for Systems and Software Engineering
hkitapci@csse.usc.edu

USC-CSSE Annual Research Review
02/13/2007
Motivation

• Major causes of software development failures
  • Lack of user involvement
  • Unrealistic expectations
  • Incomplete requirements
  • Changing requirements and specifications
• Majority of all defects in software products originate to the requirements
• The hardest single part of building a software system is deciding precisely what to build
Avoiding Requirements-Related Failures

• Make it easier for stakeholders to participate in Decision Making and Requirements Specification
  – Prototypes, brainstorming, negotiation, groupware
  – Helps getting good requirements, clarifying shared vision, managing expectations
  – Produces informal statements

• Bridge the gap between informal and more formal requirements specification; Recognized as a major challenge
Gap-Bridging Approaches

- List of Gap-bridging Approaches
  - Template-based Refinement
    - Human-initiated Template
    - Computer-initiated Template
  - Natural Language Processing
  - Keyword Analysis
  - Formal Method Experts
  - Inspections
- No approach dominated all of the other approaches
- Hybrid combinations of the approaches attractive
Research Goals

• A complete process of deriving a rigorous requirements specification from informal and unstructured statements

• Orchestrate gap-bridging techniques to provide sufficiently complete, consistent, unambiguous, and testable software requirements

• Develop an integrated support environment that brings together techniques and solutions to the issues of requirements engineering process where all stakeholders can participate in documentation and validation of requirements
Process Framework

- Change Management
- Traceability to feasible solutions
- Brainstorming
- Categorization
- Capturing terms
- Template specification
- Template filling

- Prioritization
- Conflict Identification
- Conflict Resolution
- Completeness Analysis
- Consistency Analysis
- Formal Inspections
- Informal Reviews
- Ambiguity Analysis
Hybrid Method

Requirements Negotiation
- Stakeholders
- Computer
- Reviewer
- WinWin Report
- Transformation
- Requirements Documentation
- SSRD

Natural Language Processing & Keyword Analysis
- review regulation data
- inspect WinWin report
- review requirement specification
- inspect requirements document
- SSA Report

Inspection
- use
- output

Walkthrough
- use
- output

Inspection
- do
- output

Walkthrough
- output

Inspection
- output

WinWin Report
- output

SSRD
- output
Mixed-initiated Template

• Generate report from the Negotiation results
  – Use EasyWinWin output to fill requirements templates
  – Map WinWin artifacts into Requirements template attributes
  – Final Output generated as a requirements specification document

• Modify/Update the requirement templates
  – Use extracted data to fill attributes in MARS (Measurable, Achievable, Relevant and Specific) or Use Case forms

• Customizable requirements templates and attributes
Natural Language Processing & Keyword Analysis

- Defect identification
  - Analysis to find NL defects
  - Use indicators to find problematic statement
  - Feedback to stakeholders for further correction

- Tailorable categories and checking criteria

- Extract template-relevant meanings
  - Extract verbs and nouns, priority, reference, etc.

- Feedback to stakeholders for further correction
Inspections

- Inspect results and post-process them into precise specifications
  - Use walkthroughs during the process (statement-level checking and requirement-level checking)
  - Use inspections on the outputs (document-level checking)
  - Find ambiguities, conflicting requirements, and other defects as early as possible
  - Avoid misinterpretations of WinWin artifacts
Defect Analysis on Negotiation Results

- **System navigation must be restricted to the minimum number of clicks**
  - Rule: unverifiable adjective modifier
  - Unverifiable statement
- **User authentication**
  - Rule: no subject and verb phrases
  - Unclear statement
- **Search by specific fields e.g. title, author**
  - Rule: sentence starts with verb (missing subject)
  - Missing information
- **More time is required to integrate with Z-bit or keep it as low priority**
  - Rule: proper noun
  - Unclear term
- **The database is easy to maintain and modify for the maintainer**
  - Rule: Weak phrase
  - Unverifiable statement

Frequent defect types are **missing information** and **unverifiable statement**
EasyWinWin Defect Analyzer and Transformer Agents

Natural Language Processing
- Request processing
  - Send data
    - NLP Interface
      - Display data
        - WinWinTree Formatter
          - Display data
            - WinWinTree Interface
              - Display data
                - WinWin Results
                  - Open

Defect Interface
- Display data
  - Apply rules
    - Defect Rule Editor
      - Generate
        - Reports
          - Open

Defect Rule Engine
- Request data
  - Apply attributes
    - Attribute Editor
      - Display data
        - Requirement Transformer
          - Apply attributes
            - Request data
              - Database
                - Request data
                  - Requirement Interface
                    - Request data
                      - Report Generator
                        - Generate
Conclusions

- An iterative and incremental process model with a quality assurance process
- All success-critical stakeholders participate generating requirements specification
- Semi-automated defect identification
- Semi-automated documentation and analysis of requirements specification templates
- Increase the usability of EasyWinWin process throughout the software life-cycle
Questions ?