


Integrating UCD with Requirements Engineering: Improving Processes, Formats, and Communication


Presented at UPA 2006

Lisa Battle Rebecca Ray Karen Bachmann




UCD and Requirements Engineering

- The requirements challenge
- UCD and RE have grown up in parallel
 - UCD has techniques for eliciting requirements and turning them into design
 - RE has techniques for documenting, managing and tracing requirements
 - How can we do a better job of communicating requirements?
- Background: previous workshops




Perspective #1: Rebecca Ray

- Role overlap in complex environments
 - Development teams may claim ownership of design efforts
 - In position to force their own choices because they own development effort
 - Business teams may claim ownership of design efforts
 - In position to force their own choices with management
 - Early involvement of UCD is most effective, but is HUGE challenge when you don't own requirements effort




Perspective #1: Rebecca Ray

- Progress in including UCD process and documentation in SDLC
- Presence on approval/oversight boards
- Time spent building consensus replaced with efforts focused on sharing user's story in compelling fashion (effective storytelling)
- Successes are best way for proving value, ensuring early inclusion





Perspective #2: Karen Bachmann

- Develop quick prototypes to test the requirements
- Establish a single vision of the users and tasks
- Develop usability requirements that specify how well something should work, not just what it should do (functional requirements)




Perspective #3: Lisa Battle

Which requirements are we talking about? Do we mean... ?

What the system needs to do to be successful   Numbered statements, unambiguous, testable



- “Little r” requirements
 - Are not usable
 - May be missing some of the key points
- “Big R” Requirements are a communications problem
 - Inputs (elicitation)
 - Outputs (documentation)



Your Perspective


Audience Discussion

- Are you...
 - Writing requirements?
 - Providing input to requirements?
 - Working with a requirements analyst and noticing overlap between your role and theirs?
 - Serving as the requirements analyst?
- What challenges are you facing?



Issues for Discussion


- Coordination of roles and activities
- Integrating artifacts produced by UCD and RE
- Effective formats for communicating requirements
- Getting the essentials into the requirements
- Usability as "non-functional" requirement
- Goals for the future



Roles and Activities of UCD and RE in Software Development

Background


- Usability professionals increasingly lead requirements elicitation and documentation
- Role overlap occurs between usability professionals and others



Roles and Activities of UCD and RE in Software Development

Questions

- How can we coordinate activities to avoid bottlenecks and duplication of work?
- What should be the “touch points” between UCD and RE processes?

[Karen](#) [Lisa](#) [Rebecca](#) 

Roles and Activities of UCD and RE in Software Development

Panelist Recommendations: Karen


- Establish a respectful relationship
 - Differences in background
 - Differences in approach and focus
 - Same end goal: A successful product and happy users
- Share the requirements gathering tasks based on the strengths of each discipline – strong project planning!
- Recognize a shared pain: Many development organizations do not appreciate the value of requirements either

Roles and Activities of UCD and RE in Software Development

Panelist Recommendations: Lisa

UCD	RE
Figure out what the real needs are.	Elicitation of requirements
Design a solution to meet the needs.	Analysis of requirements
Make sure it works.	Documentation of requirements
	Tracking/management of requirements


UCD is a proven method for eliciting and validating requirements



Roles and Activities of UCD and RE in Software Development

Panelist Recommendations: Rebecca


- Close alignment with work initiation process
 - Begin user research early enough to influence but not hinder
- Formal UCD checkpoints in SDLC
- UCD membership on approval boards, oversight committees
- Active involvement for business analysts:
 - UCD Approach to Develop Effective Business Requirements (class/workshop)
 - Shift/share responsibility for basic UCD activities where appropriate




Roles and Activities of UCD and RE in Software Development

Audience Discussion

- How can we coordinate activities to avoid bottlenecks and duplication of work?
- What should be the “touch points” between UCD and RE processes?






Integrating Artifacts Produced by UCD and RE

Background


- Many types of artifacts are produced, including:
 - RE: Business requirements, user requirements, business rules, functional requirements, activity diagrams, use cases
 - UCD: Personas, scenarios, user interface standards and style guides, low-fidelity and high-fidelity prototypes, usability goals, usability test findings
- Comparison of artifacts from different disciplines
- Relationship between artifacts




Integrating Artifacts Produced by UCD and RE

Questions

- How well do UCD artifacts feed into the requirements documents and other artifacts produced in software development?
- Should we integrate UCD deliverables with other systems requirements documents?

[Karen](#) [Rebecca](#) [Lisa](#) 






Integrating Artifacts Produced by UCD and RE

Panelist Recommendations: Karen

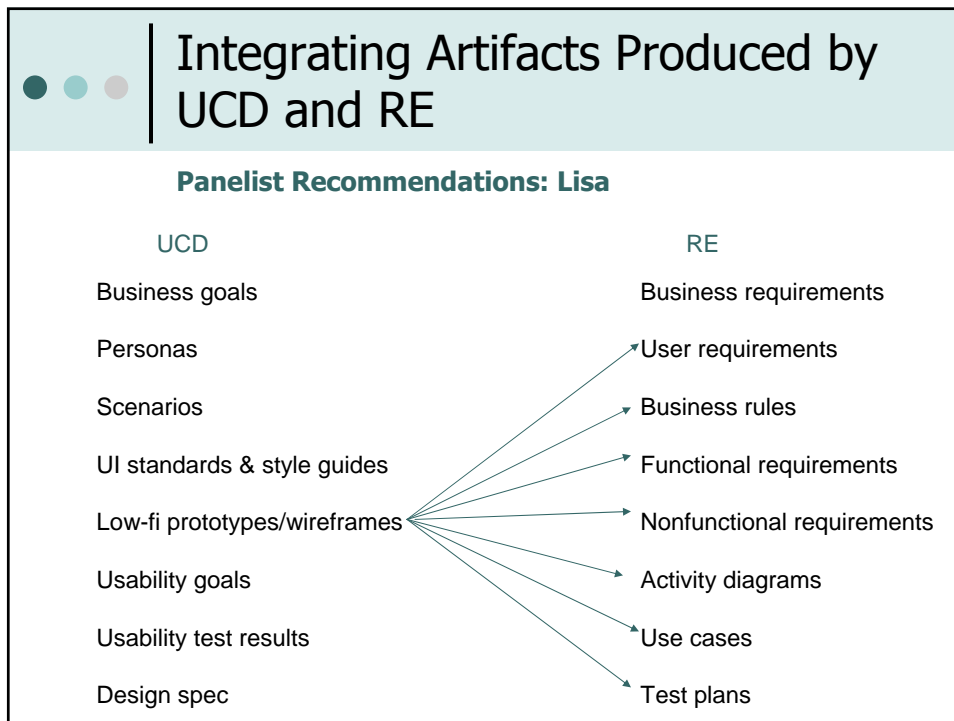
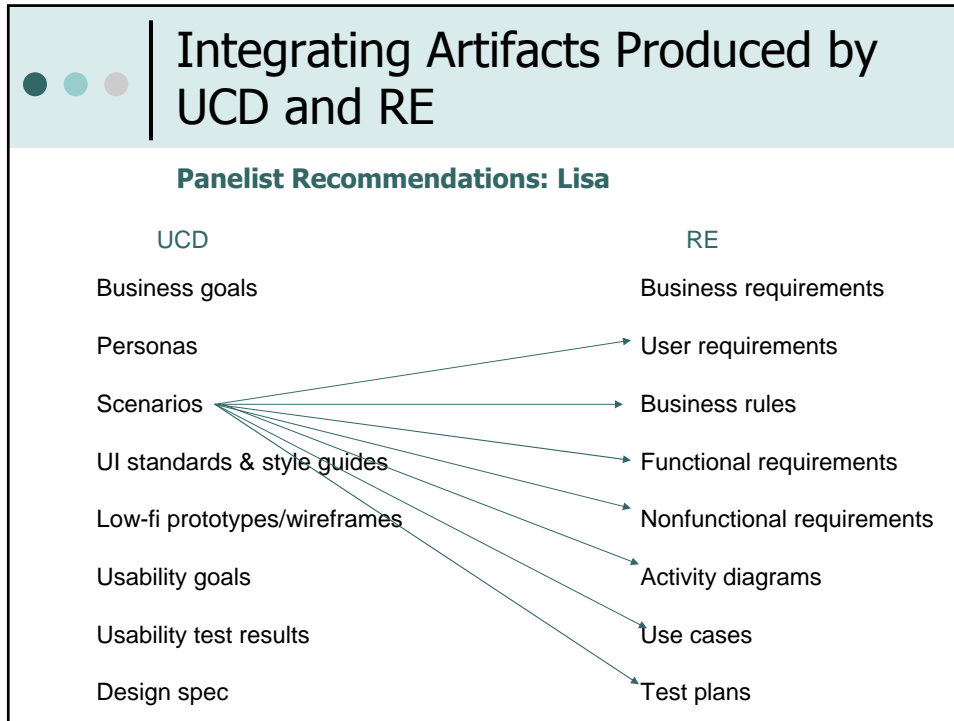
- Identify where each deliverable fits in the development life cycle and how each relates
- Always strive to trace back to the user throughout the process and with every artifact




Integrating Artifacts Produced by UCD and RE

Panelist Recommendations: Rebecca

- Good fits for existing documents:
 - objectives and goals
 - usability requirements
 - prototypes
- Clarification and coordination needed when BA or development team own use cases
- Timely presentation of user research findings and usability testing results can greatly influence direction and decisions







Integrating Artifacts Produced by UCD and RE

Audience Discussion

- How well do UCD artifacts feed into the requirements documents and other artifacts produced in software development?
- Should we integrate UCD deliverables with other systems requirements documents?



Effective Formats for Communicating Requirements

Background

- Problems with traditional formats for requirements documentation
- Problems with UCD artifacts
- Making artifacts more useful for communicating:
 - A vision to team and stakeholders
 - Concise but detailed information to developers
 - Appropriate types of information depending on project context

Effective Formats for Communicating Requirements

Questions

- What formats work best for documenting each type of requirement?
- Do UCD artifacts communicate well to engineers and software developers?

[Lisa](#) [Rebecca](#) [Karen](#)

Effective Formats for Communicating Requirements

Panelist Recommendations: Lisa

5.1 Normal Flow

5.1.1 This use case starts when the Proposal Information Administrator requests to administer proposal information.

5.1.2 System requests proposal information

5.1.3 Proposal information administrator provides proposal information.

5.1.4 System requests a decision to submit proposal information.

5.1.5 Proposal Information Administrator provides proposal information submission decision.

5.1.6 Proposal Information Administrator submits proposal information.

5.1.7 System validates submitted proposal information.

5.1.8 System saves submitted proposal information.

5.1.9 System requests decision to update proposal information

5.2 Alter Requirements Artifacts

5.2.1 View or update proposal information

5.2.1.1 System generates proposal identification information.

5.2.1.2 System provides proposal identification information for selection.

5.2.1.3 System requests decision to update proposal identification information display perspective.

5.2.1.4 System requests decision to update proposal identification

Effective Formats for Communicating Requirements

Panelist Recommendations: Lisa

5.1 Normal Flow

5.1.1 This use case starts when the Proposal Information Administrator requests to administer proposal information.

5.1.2 System requests proposal information

5.1.3 Proposal information administrator provides proposal information.

5.1.4 System requests a decision to submit proposal information.

5.1.5 Proposal Information Administrator provides proposal submission decision.

5.1.6 Proposal Information Administrator submits proposal information.

5.1.7 System validates submitted proposal information.

5.1.8 System saves submitted proposal information.

5.1.9 System requests decision to update proposal information display perspective.

5.2 Alter

5.2.1 View or update proposal information

5.2.1.1 System generates proposal identification information.

5.2.1.2 System provides proposal identification information for selection.

5.2.1.3 System requests decision to update proposal identification information display perspective.

5.2.1.4 System requests decision to update proposal identification

Typical problems:

- Too big and complex
- Does not adequately reflect the reasons, which are grounded in user-centered analysis
- Stakeholders and users cannot tell from reading them whether or not the requirements reflect what they wanted

Requirements Artifacts


Effective Formats for Communicating Requirements

Panelist Recommendations: Lisa

Typical problems:

- Not considered detailed and formal enough (not “sign-off worthy”)
- May not communicate all of the intended behaviors and business logic

UCD Artifacts




Effective Formats for Communicating Requirements

Panelist Recommendations: Lisa

Recommendations:


- Adopt a “living” format that can be updated
- Create easily digestible, small, granular artifacts
- Cross-reference between artifacts (avoid introducing redundancy or possible inconsistencies)
- Form clusters of UX requirements into vignettes around scenarios and usage goals
 - Example: link scenario with related use case(s), user profile(s), and wireframes
- Use matrices and data tables to describe requirements for adaptive or data driven UIs



Effective Formats for Communicating Requirements

Panelist Recommendations: Rebecca


- Usability tests with audio and video
- Heuristic reviews most effective when robust industry research included
 - Not opinion but proven fact
- Partnership/consultation during prototype creation
 - Especially when ownership issues exist



Effective Formats for Communicating Requirements

Panelist Recommendations: Karen


- Communicate in the vernacular most likely to succeed for the project and organization
 - May not look like traditional UCD deliverables
 - Should look “familiar” to other team members
- Integrate UCD into established requirements deliverables – if they exist




Effective Formats for Communicating Requirements

Audience Discussion

- What formats work best for documenting each type of requirement?
- Do UCD artifacts communicate well to engineers and software developers?






Getting the Essentials into the Requirements

Background


- There is a risk that the results of user-centered activities are not translated into requirements




Getting the Essentials into the Requirements

Questions

- How can we make sure that UCD findings are translated into well-written requirements?
 - Usability test results
 - User observation, interviews, and contextual inquiry findings
 - Heuristic evaluations
- Can we quantify user requirements in a testable way?


[Karen](#) [Rebecca](#) [Lisa](#) 



Getting the Essentials into the Requirements

Panelist Recommendations: Karen


- Constructing usability requirements:
 - Determine what usability criteria to measure and the priority for each
 - Determine how the criteria be measured: Create **tangible** measurements of intangible user satisfaction statements
 - Set a realistic percentage of users that must achieve the goals
 - Define the conditions that must exist for the product to successfully fulfill the requirements



Getting the Essentials into the Requirements

Panelist Recommendations: Karen


- Components of a usability requirement
 - What task should the user accomplish?
 - Who will accomplish the task?
 - What conditions will the task be performed under?
 - How well should the task be performed?



Getting the Essentials into the Requirements

Panelist Recommendations: Karen

- General tips
 - Convert qualitative wants and needs to quantifiable goals (absolute v. relative)
 - Write them in terms of user tasks and goals
 - Prioritize needs of different user groups
 - Prioritize the usability requirements
 - Be realistic – success is rarely 100% of users
 - Test the requirements



Getting the Essentials into the Requirements

Panelist Recommendations: Rebecca

- Present research and test results in compelling formats, with concrete recommendations, with calls to action
- Use audio and video wherever possible - written reports sometimes easier to file away and ignore
- Expose results (where politically appropriate) to influential audience
- Partnership/consultation during prototype creation can help to introduce important requirements that may have been missed

Getting the Essentials into the Requirements

Panelist Recommendations: Lisa

<ul style="list-style-type: none"> Job Experience Vocabulary Expectations Roles Motivations 	<ul style="list-style-type: none"> Education Mental Models Common Misconceptions Priorities Likes and Dislikes 	<ul style="list-style-type: none"> Opportunities for Improvement Business Process/Workflow Business Drivers Political Issues Incentives Problems in the Current Process Physical Work Environment Organizational Structure Stakeholders Other automated systems in use
--	---	--


- Triggering Events
- Sequence of Steps
- Relationship to other Tasks
- Inputs & Outputs
- Success Criteria
- Common Errors

Do our requirements trace back to all of these things we learned?

Getting the Essentials into the Requirements

Panelist Recommendations: Lisa



- Multiple types of requirements may be generated based on a single UCD finding (see handout)



Getting the Essentials into the Requirements

Audience Discussion


- How can we make sure that UCD findings are translated into well-written requirements?
 - Usability test results
 - User observation, interviews, and contextual inquiry findings
 - Heuristic evaluations
- Can we quantify user requirements in a testable way?



Usability as a “Non-functional” Req

Background


- Categories of requirements typically include “functional” and “non-functional” (and sometimes others)
- Usability is in the “non-functional” category
- Non-functional requirements are perceived as less important
- UCD activities elicit all types of requirements, not just usability requirements



Usability as a “Nonfunctional” Req

Questions


- Is this a problem?
- If so, what can we do to change it?




Usability as a “Nonfunctional” Req

Audience Discussion

- Is this a problem?
- If so, what can we do to change it?






Goals for the Future

Questions

- What do we want our role to be in relation to requirements engineering?
- How can we position ourselves to take on that role in the future?


[Lisa](#) [Rebecca](#) [Karen](#) ▶



Goals for the Future

Panelist Recommendations: Lisa


- Lead user-centered elicitation
- Promote iterative prototyping and user feedback
- Invent more effective communication formats
- Ownership of the documentation as appropriate for the organization



Goals for the Future

Panelist Recommendations: Rebecca


- Influence early with compelling voice of customer data
- Do not try to own the requirements effort
 - UCD service area is overwhelmingly large
- Maybe we should opt to remain in totally unbiased role – point of discussion



Goals for the Future

Panelist Recommendations: Karen



- Our role:
 - Ingrain a user-focus that starts with the requirements and continues to the final delivery
 - Share tasks and build on the strengths of each discipline to increase efficiency and effectiveness in supporting development
- How to get there:
 - Educate ourselves about requirements engineering
 - Reach out to RAs based on increased understanding of their work



Goals for the Future

Audience Discussion

- What do we want our role to be in relation to requirements engineering?
- How can we position ourselves to take on that role in the future?



Questions and Discussion

Lisa Battle
lbattle@acm.org

Rebecca Ray
rebecca.p.ray@gsk.com

Karen Bachmann
karen@seaconinc.com