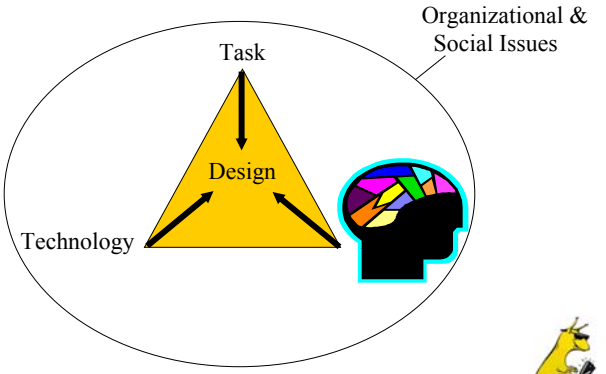
HCI Application: Inspection and Testing



Different way of looking at evaluation

- Inquiry: involving real users but not in a usability laboratory
  - "ecological validity" of evaluation
- Sample methods
  - Proactive field study: before starting the design, the usability expert talks to users
  - Field/ethnographic observation
  - Focus Groups
  - Interviews & contextual inquiry
  - Logging Actual Use
  - Questionnaires

Different way of looking at evaluation

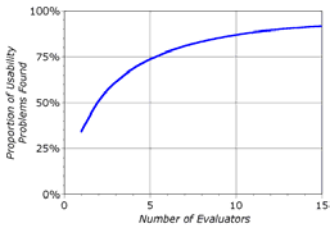
- Inspection: usability experts make judgements
  - Based on experience
  - Guided by rules of thumb, guidelines, psychological theories/models
- Rationale:
  - Observing users can be time-consuming and expensive
  - Try to predict usability rather than observing it directly
  - Conserve resources (quick & low cost)
- Expert reviewers often used
  - HCI experts interact with system and try to find potential problems and give prescriptive feedback
  - Best if they
    - Haven't used earlier prototype
    - Familiar with domain or task
    - Understand user perspectives

Predictive/Expert Evaluation/Inspection

- Sample methods:
  - Perspective-based Inspection: inspecting the system in 3 scenarios: expert users, novice users and error handling. Each inspector only works on 1 scenario
  - Feature Inspection: System features are inspected in the context of a task (e.g. word processor in the context of writing a letter).
  - Heuristic evaluation → using rules of thumbs to evaluate user interface in terms of their violation severity
  - Cognitive/pluralistic walkthroughs
  - User/predictive modelling

Heuristic Evaluation

- Most famous: Nielsen's heuristics by Jakob Nielsen & Rolf Molich
- Several expert usability evaluators assess system based on simple and general heuristics (principles or rules of thumb) independently
- Famous quote: 5 is more than enough



http://www.useit.com/

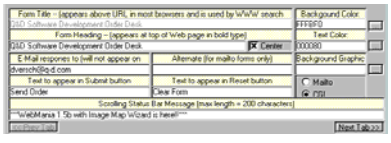
http://www.dialogdesign.dk/inenglish.html

Nielsen's Heuristics

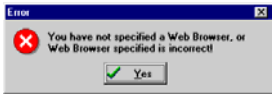
- Recognition rather than recall

- Flexibility and efficiency of use

  - Shortcut keys
- Aesthetic and minimalist design

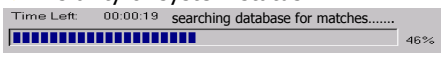


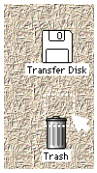
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Nielsen's Heuristics

- Visibility of system status





- Match between system and the real world
- User control and freedom

- Consistency and standards

- Error prevention

Nielsen's Heuristics for Websites

- Visibility of system status: where am I? where can I go next? Breadcrumb
- Match between system and the real world: site structure maps real-world structure
- User control and freedom: relative sizes
- Consistency and standards: site naming, HTML conventions. "Shipping under Help"
- Error prevention: especially in forms
- Recognition rather than recall: tool tips, visited vs. unvisited links, etc
- Flexibility and efficiency of use: skip intro, jump to, search
- Aesthetic and minimalist design: white space, progressive level of details
- Help users recognize, diagnose, and recover from errors: error messages (again in forms)
- Help and documentation

http://instone.org/heuristics

Procedure

1. Gather inputs
   - Who are evaluators?
     - Need to learn about domain, its practices
   - Get the prototype to be studied
     - May vary from mock-ups and storyboards to a working system
2. Evaluate system
3. Debriefing and collection
4. Severity rating

2: Evaluate System

- Reviewers evaluate system based on high-level heuristics.
- Perform two or more passes through system inspecting
  - Each screen
  - Flow from screen to screen
- Evaluate against heuristics
- Find "problems"
  - Subjective (if you think it is, it is)
  - Don't dwell on whether it is or isn't

3: Debriefing

- Organize all problems found by different reviewers
  - At this point, decide what are and aren't problems
  - Group, structure
  - Document and record them

4: Severity Rating

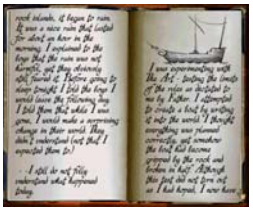
- 0-4 rating scale
- Based on
  - frequency
  - impact
  - persistence
  - market impact

Heuristics for Virtual Environment

1. Natural engagement: user should be unaware that the reality is virtual
2. Realistic feedback: the effect of the user's actions on virtual world objects should be immediately visible and conform to the laws of physics
3. Clear entry and exit points: clear indication on how to enter and exit from a virtual world
4. Clear turn-taking: where system initiative is used it should be clearly signaled and follow conventions
5. Navigation and orientation support: the users should always be able to find where they are in the VE and return to known, preset positions
6. Faithful viewpoints: the visual representation of the virtual world should map to the user's normal perception
7. Support for learning: active objects should be cued and if necessary explain themselves to promote learning of VEs

http://www.informatics.manchester.ac.uk/hci_design/HeurCompleteVn2.doc

Games Heuristics



- Minimize flashing
- Avoid large blocks of text
- Don't rely on player's memory
  - Don't use acronyms/abbreviations
  - Don't ask players to count resources
  - Don't ask players to remember level design
- Display only relevant information
- Don't bury frequently used information

Games Heuristics

- Make critical information stand out
- Provide contextual information (e.g. where they are in mini-map)
- Players should understand goals, failures, game elements (enemies, avatars, obstacles)
- Provide control (room for errors, moving to the next level)

Heuristics for Ambient Display

- Useful and relevant information
- "Peripherality" of display
  - Unobtrusive but easily monitor-able
- Match between design of ambient display and environments
  - Display should be noticed for its data change rather than clash with environment
- Sufficient information design
- Consistent and intuitive mapping
- Easy transition to more in-depth information
  - For multi-level information, ease of switching between focus and context
- Visibility of state
- Aesthetic and pleasing design

http://doi.acm.org/10.1145/642611.642642

Trade-offs

- Advantages
  - Few ethical issues to consider
  - Inexpensive, quick
  - Getting someone practiced in method and knowledgeable of domain is valuable
  - Talking the same language
- Challenges
  - Very subjective assessment of problems - depends of expertise of reviewers
  - Which heuristics?
  - How to determine what is a true usability problem

Cognitive Walkthrough

- Walkthrough the interface based on a cognitive model
- Evaluation of actions and cues of the interface in comparison to the goals and the background of the typical users
- Like code walkthrough (s/w engineering)
- Pluralistic walkthrough – like CW but done by a pair
- Process:
  - Construct carefully designed tasks from system spec or screen mock-up
  - Walk through (cognitive & operational) activities required to go from one screen to another
  - Review actions needed for task, attempt to predict how users would behave and what problems they'll encounter

Requirements and Assumptions

- Requirements:
  - Description of users and their backgrounds
  - Description of task user is to perform
  - Complete list of the actions required to complete task
  - Prototype or description of system
- Assumptions
  - User has rough plan
  - User explores system, looking for actions to contribute to performance of action
  - User selects action seems best for desired goal
  - User interprets response and assesses whether progress has been made toward completing task

CW: Methodology

- Preparation: describing user profile, choosing tasks, breaking down tasks
- Evaluation: answering 4 questions by creating success story or failure story :
  1. Will the user try to achieve right effect? (user thought at the beginning of the action)
  2. Will the user notice that the correct action is available? (user ability to locate the order)
  3. Will the user associate the correct action with the effect that user is trying to achieve? (user ability to identify the control)
  4. If the correct action is performed, will the user see that progress is being made toward solution of the task? (user ability to interpret the information feedback)

CW: Answering the Questions

- 1. Will user be trying to produce right effect?
  - Typical supporting evidence
    - It is part of their original task
    - They have experience using the system
    - The system tells them to do it
- 2.Will user notice that correct action is available?
  - Typical supporting evidence
    - Experience
    - Visible device, such as a button
    - Perceivable representation of an action such as a menu item

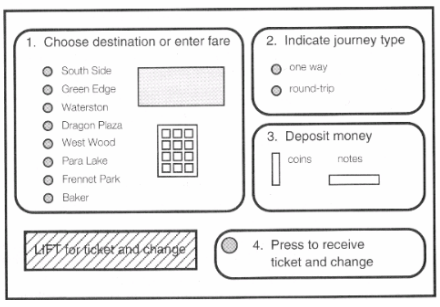
CW: Next Question

- 3.Will user know it's the right one for the effect?
  - Typical supporting evidence
    - Experience
    - Interface provides a visual item (such as prompt) to connect action to result effect
    - All other actions look wrong
- 4.Will user understand the feedback?
  - Typical supporting evidence
    - Experience
    - Recognize a connection between a system response and what user was trying to do

CW: Ticket machine

Task: buying RT ticket to Dragon Plaza

- Q1: Does user know how many tickets are produced?



- Q2: Does user know the sequence?
- Q3: Does user know how to select?
- Q4: Does user know that s/he has chosen Dragon Plaza RT? Does user know whether the right amount of money had been put in?

Predictive Modeling

- Idea: If we can build a model of how a user works, then we can predict how s/he will interact with the interface
  - Predictive model → predictive evaluation
- No mock-ups or prototypes!
- Stimulus-Response
  - Hick's law
  - Practice law
  - Fitt's law
- Cognitive – human as interperter/predictor – based on Model Human Processor (MHP)
  - Key-stroke Level Model
    - Low-level, simple
  - GOMS (and similar cognitive) Models
    - Higher-level (Goals, Operations, Methods, Selections)

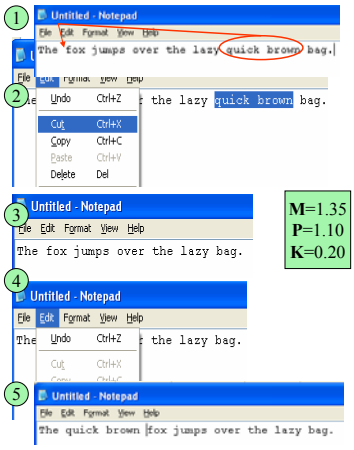
Power law of practice

- $T_n = T_1 n^{-a}$
  - T_n to complete the nth trial is T_1 on the first trial times n to the power -a; a is about .4, between .2 and .6
  - Skilled behavior - Stimulus-Response and routine cognitive actions
    - Typing speed improvement
    - Learning to use mouse
    - Pushing buttons in response to stimuli
    - NOT learning
- Use measured time T_1 on trial 1 to predict whether time with practice will meet usability criteria, after a reasonable number of trials
  - How many trials are reasonable?
- Predict how many practices will be needed for user to meet usability criteria
  - Determine if usability criteria is realistic

Hick's law

- Decision time to choose among n equally likely alternatives
  - $T = I_c \log_2(n+1)$
  - $I_c \sim 150$ msec
- Menu selection
- Which will be faster as way to choose from 64 choices?
  - Single menu of 64 items
  - Two-level menu of 8 choices at each level
  - Two-level menu of 4 and then 16 choices
  - Two-level menu of 16 and then 4 choices
  - Three-level menu of 4 choices at each level
  - Binary menu with 6 levels

KLM: Method Used
Cut-and-paste-using-menus



M=1.35
P=1.10
K=0.20

Description	Opr	Dur (s)
Mentally Prepare	M	1.35
Move cursor to "quick"	P	1.10
Double-click mouse button	K	0.40
Move cursor to "brown"	P	1.10
Shift-click mouse button	K	0.40
Mentally Prepare	M	1.35
Move cursor to Edit menu	P	1.10
Click mouse button	K	0.20
Move cursor to Cut menu	P	1.10
Click mouse button	K	0.20
Mentally Prepare	M	1.35
Move cursor to before "fox"	P	1.10
Click mouse button	K	0.20
Mentally Prepare	M	1.35
Move cursor to Edit menu	P	1.10
Click mouse button	K	0.20
Move cursor to Paste menu item	P	1.10
Click mouse button	K	0.20
TOTAL PREDICTED TIME		**14.90**

Different way of looking at evaluation

- Testing: involving users directly in usability laboratory tests
- Sample methods
  - Coaching Method
  - Co-discovery Learning
  - Performance Measurement
  - Question-asking Protocol
  - Remote Testing
  - Retrospective Testing
  - Shadowing Method
  - Teaching Method
  - Thinking Aloud Protocol

Usability Evaluation Testing

- Thinking Aloud Protocol
  - Users are asked to verbalize their thoughts, feelings and opinions when using the system.
  - Two variations: critical response & periodic report.
  - Useful to understand user's mental model, interaction with the system and terminology.
- Co-discovery Learning
  - Users work in pair to achieve a common goal with the tested system.
  - Users are encouraged to "think-aloud".
- Shadowing method
  - An expert user sits next to a user and explains the user's behaviour to the usability tester.
  - Usually when it is not appropriate for users to give any response during the test session.

Usability Evaluation Testing

- Coaching method
  - Participants asked an expert coach.
  - Participant-coach & participant-computer interactions are observed.
- Teaching method
  - User interacts with the system to gain some expertise.
  - A novice user is brought in, and the "expert" user is asked to explain to the novice user how to perform the task.
- Performance measurement/controlled experiment
  - Obtaining quantitative data when participants perform a task.
  - Minimize participant-tester interaction during the test as it might affect the quality of data.

Usability Evaluation Testing

- Question-asking Protocol
  - Tester asked questions directly to participants.
  - Participants asked in terms of their past experience of in relation to the tested system.
- Remote Testing
  - Tester are separated in time/space from users.
  - Data are obtained from logs/records/networks.
  - Combined with other methods
- Retrospective Testing
  - Tester & participants reviewed recorded session together, tester asked what was going on.
  - Combined with other methods.

Controlled Experiments

- Design the experiment to collect the data to test the hypotheses to evaluate the interface to refine the design
- Three elements: *quantitative, experimental,* with *end users*.
- Benchmark tasks - gather quantitative data
  - Specific, clearly stated task for users to carry out
  - Example: "Find the message from Mary and reply with a response of 'Tuesday morning at 11'."
  - Users perform these under a variety of conditions and you measure performance (time, errors, etc).
- Representative tasks - add breadth, can help understand process

Variables

- "independent" = the things you compare (e.g. using mouse or joystick)
- "dependent" = the things you observe and measure (e.g. time taken to navigate)
- "control" = the things you don't want to interfere (e.g. age, order of trying each alternative)
  - Don't allow it to vary: e.g., all males
  - Allow it to vary randomly: e.g., randomly assign participants to different groups
  - Counterbalance - systematically vary it: e.g., equal number of males, females in each group
- "nuisance" = the things you forgot to control!

Hypotheses

- What you predict will happen
- More specifically, the way you predict the dependent variable (i.e., accuracy) will depend on the independent variable(s)
- "Null" hypothesis (H_o)
  - Stating that there will be no effect
  - e.g., "There will be no difference in performance between the two groups"
  - Data used to try to disprove this null hypothesis

Example

- Do people complete operations faster with a black-and-white display or a color one?
  - Independent - display type (color or b/w)
  - Dependent - time to complete task (minutes)
  - Control - same number of novices and experts in each group
  - Hypothesis: Time to complete the task will be shorter for users with color display
  - H_o: $Time_{color} = Time_{b/w}$

Experimental Designs

- Within Subjects Design
  - Every participant provides a score for all levels or conditions

	Color	B/W
P1	12 secs.	17 secs.
P2	19 secs.	15 secs.
P3	13 secs.	21 secs.
…		

- Between Subjects
  - Each participant provides results for only one condition

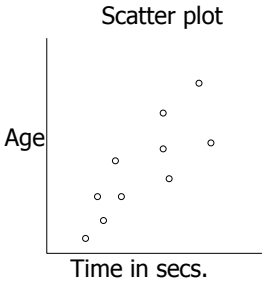
Color		B/W	
P1	12 secs.	P2	17 secs.
P3	19 secs.	P5	15 secs.
P4	13 secs.	P6	21 secs.
…			

Trade-offs

- Within-subject
  - More efficient: Each subject gives you more data - they complete more "blocks" or "sessions"
  - More statistical "power": Each person is their own control
  - Therefore, can require fewer participants
  - May mean more complicated design to avoid "order effects": e.g. participants may learn from first condition or fatigued after the first condition
- Between subject
  - Fewer order effects
  - Simpler design & analysis
  - Easier to recruit participants (only one session)
  - Less efficient

What about subjects?

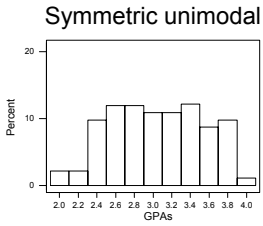
- How many?
  - Book advice:at least 10
  - Other advice:6 subjects per experimental condition
  - Real advice: depends on statistics
- How do you know you had enough?
  - First look at each participant's data
  - Were there data that are very different from the rest, people who fell asleep, anyone who tried to mess up the study, etc.?
  - Then look at aggregate results, descriptive statistics or plot

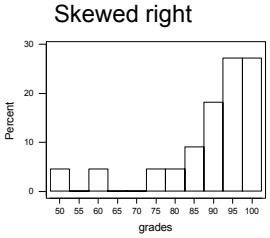


Descriptive Statistics

- Max, min
- **Measures of location**. Ex. Data: 3 8 3 3 1
  - A variable that is way different than others is called an "**outlier**" (=8).
  - Mean (average) = (3+8+3+3+1)/5 = 3.6 → affected by outliers
  - Median = the middle point of sequentially arranged data points (1,3,3,3,8) = 3 → Robust to outliers
  - Mode: the most frequently showing data = 3 → Robust to outliers
  - The most appropriate measure depends on distribution shape (symmetric vs. skewed, unimodal vs. multimodal)

Data distribution shapes

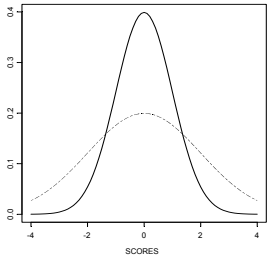




- If data are symmetric, the mean, median, and mode will be approximately the same.
- If data are multimodal, report the mean, median and/or mode **for each subgroup**.
- If data are skewed, report the median.

2.4.3 Measures of Spread

- What is spread or dispersion? The degree to which scores are clumped around the mean.
  - Standard deviation: The square root of the average squared deviation score
  - Variance: The average squared deviation score.



$$SD = \sigma = \sqrt{\frac{\sum (x - M)^2}{N}}$$

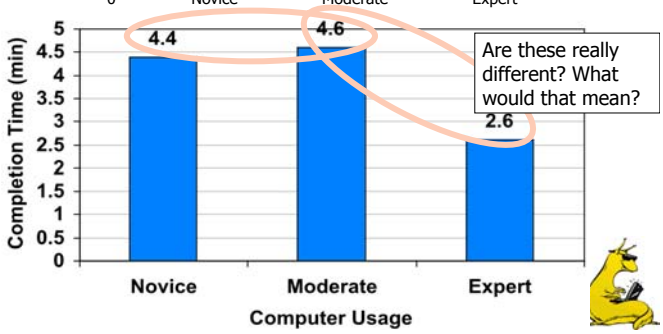
$$VAR = \sigma^2 = \frac{\sum (x - M)^2}{N}$$

Experimental Results

- How does one know if an experiment's results mean anything or confirm any beliefs?
- Example: 40 people participated,
  28 preferred interface 1,
  12 preferred interface 2
- Inferential Statistics
  - Tests to determine if what you see in the data (e.g., differences in the means) are reliable (replicable), and if they are likely caused by the independent variables, and not due to random effects
  - e.g. t-test to compare two means
  - e.g. ANOVA (Analysis of Variance) to compare several means
  - e.g. test "significance level" of a correlation between two variables

Inferential Stats and the Data

- Ask diagnostic questions about the data
- Recall: We set up a "null hypothesis"
  - e.g. there should be no difference between the completion times of the three groups
  - Or H_0: $Time_{Novice} = Time_{Moderate} = Time_{Expert}$



Hypothesis Testing

- "Significance level" (p):
  - The probability that your null hypothesis was wrong, simply by chance
  - Can also think of this as the probability that your "real" hypothesis (not the null), is wrong
  - The cutoff or threshold level of p ("alpha" level) is often set at 0.05, or 5% of the time you'll get the result you saw, just by chance
  - e.g. If your statistical t-test (testing the difference between two means) returns a t-value of t=4.5, and a p-value of p=.01, the difference between the means is statistically significant

Drawing Conclusions

- Errors in analysis do occur – main types:
  - Type I/False positive - You conclude there is a difference, when in fact there isn't
  - Type II/False negative - You conclude there is no difference when there is
- Make your conclusions based on the descriptive stats, but back them up with inferential stats
  - e.g., "The expert group performed faster than the novice group $t_{(1,34)} = 4.6$, $p \leq .05$."
- Translate the stats into words that regular people can understand
  - e.g., "Thus, those who have computer experience will be able to perform better, right from the beginning…"

Feeding Back Into Design

- What were the conclusions you reached?
- How can you improve on the design?
- What are quantitative benefits of the redesign?
  - e.g. 2 minutes saved per transaction, which means 24% increase in production, or $45,000,000 per year in increased profit
- What are qualitative, less tangible benefit(s)?
  - e.g. workers will be less bored, less tired, and therefore more interested → better customer service
- Compare the learnability of concrete vs. abstract icons
  - Hypothesis? Variables? Experimental design? Tasks?

Summary: Method vs. Stage

Evaluation Methods	Stages in Software Development Life-cycle				
	Requirements	Design	Code	Test	Deploy-ment
Proactive Field Study	*	*			
Pluralistic Walkthroughs		*			
Shadowing Method		*	*	*	
Heuristic Evaluation		*	*	*	*
Cognitive Walkthroughs		*	*	*	*
Coaching Method		*	*	*	*
Performance Measurement		*	*	*	*
Think-aloud Protocol		*	*	*	*

Summary: Method vs. Stage

Evaluation Methods	Stages in Software Development Life-cycle				
	Requirement	Design	Code	Test	Deployment
Interviews		*	*	*	*
Retrospective Testing		*	*	*	*
Feature Inspection			*	*	*
Focus Groups				*	*
Questionnaires				*	*
Field Observation				*	*
Logging Actual Use				*	*

Costs

Low Cost:
- Heuristic Evaluation

Medium Cost:
- Cognitive Walkthroughs
- Field Observation
- Interviews
- Logging Actual Use
- Proactive Field Study
- Questionnaires

High Cost:
- Coaching Method
- Focus Groups
- Performance Measurement
- Pluralistic Walkthroughs
- Question-asking Protocol
- Retrospective Testing
- Shadowing Method
- Thinking-aloud Protocol

Summary: Method vs. Resources

Characteristics		Evaluation Methods	Proactive Field Study	Pluralistic Walkthroughs	Teaching Method	Shadowing Method
Personnel		Usability experts	1	1	1	1
		Software developers	0	1	0	0
		Users	2	2	4	4
Remote Testing			No	No	No	No
Quantitative Data			No	No	No	Yes
Usability Issues		Effectiveness	No	Yes	Yes	Yes
		Efficiency	No	No	No	Yes
		Satisfaction	No	Yes	Yes	No

Summary: Method vs. Resources

Characteristics		Evaluation Methods	Co-discovery Learning	Question-asking Protocol	Scenario-based Checklists	Heuristic Evaluation
Personnel		Usability experts	1	1	1-3	4
		Software developers	0	0	0	0
		Users	6	4	0	0
Remote Testing			No	No	Yes	Yes
Quantitative Data			No	No	No	No
Usability Issues		Effectiveness	Yes	Yes	Yes	Yes
		Efficiency	No	No	Yes	Yes
		Satisfaction	Yes	Yes	No	No

Summary: Method vs. Resources

Characteristics		Evaluation Methods	Thinking-aloud Protocol	Cognitive Walkthrough	Coaching Method	Performance Measurement
Personnel		Usability experts	1	1-4	1	1
		Software developers	0	0-2	0	0
		Users	4	0	4	6
Remote Testing			No	No	No	No
Quantitative Data			No	No	No	Yes
Usability Issues		Effectiveness	Yes	Yes	Yes	Yes
		Efficiency	No	No	No	Yes
		Satisfaction	Yes	No	Yes	No

Summary: Method vs. Resources

Characteristics		Evaluation Methods	Interviews *	Retrospective Testing	Remote Testing	Feature Inspection
Personnel		Usability experts	1	1	1	1
		Software developers	0	0	0	0
		Users	2	4	5	0
Remote Testing			Yes	No	Yes	Yes
Quantitative Data			Yes	Yes	Yes	No
Usability Issues		Effectiveness	Yes	Yes	Yes	Yes
		Efficiency	Yes	Yes	Yes	No
		Satisfaction	Yes	Yes	Yes	No

Summary: Method vs. Resources

Characteristics		Evaluation Methods	Focus Groups	Questionnaires	Field Observation	Logging Actual Use
Personnel		Usability experts	1	1	1	1
		Software developers	0	0	0	0
		Users	6	6	2	6
Remote Testing			No	Yes	No	Yes
Quantitative Data			No	Yes	No	No
Usability Issues		Effectiveness	Yes	Yes	Yes	Yes
		Efficiency	No	Yes	No	Yes
		Satisfaction	Yes	Yes	Yes	No