Cognitive Walkthroughs

by Clayton Lewis and Cathleen Wharton

presented by Jochen Rode

Outline

• Method
• How to
• Advantages&disadvantages
• Related methods
• Conclusions
Method

- evaluate UI by analyzing the mental processes required of users
- developed in 1990, 3rd refined version in 1992
- Cognitive Walkthrough’s (CW) key features:
  - identifies problems in early designs
  - performed by analyst/group of analysts (vs. user)
  - examines specific user tasks (vs. whole interface)
  - analyzes correct sequences of action

How to - Preparation

(1) define assumed user background
  - computer literacy (MS Windows, Mac Apps)
  - task knowledge
  - anticipated use environment (kitchen, cockpit,…)
(2) choose specific, realistic task
  - evaluation usually requires several CWs
  - important task (frequent or infrequent but critical)
  - realistic task (not isolated test of system features)
How to - Preparation (cont’d)

(3) specify correct action sequence
   – analyst specifies correct way of doing the task
     (the path the designer wants the user to follow)

(4) determine interface states along sequence
   – What does the user "see" at each step?
   – What are the user’s options?

How to - Analysis

(1) examine sequences for problems
   – trace the likely mental processes of the user
   – Will the user be trying to achieve the right effect?
   – Will the user notice that the correct action is available?
   – Will the user associate the correct action with the desired effect?
   – If the correct action is performed, will the user see that progress is being made?
How to - Analysis (cont’d)

(2) construct success or failure story
   – Why would the user choose/not choose an option?
   – record problems, reasons, and assumptions
(3) consider and record design alternatives
   – especially appropriate when analyst is designer

Follow-up: - modify design

Advantages

• early design evaluation (no prototype needed)
• little effort
  – no user testing
  – designer can play the role of the analyst
• finds about 40% of the problems revealed by user testing
• does not require learning of a modeling framework (vs. GOMS for example)
Disadvantages & Dangers

- not all interface features are covered
- choosing a good task is difficult
  - temptation to improve coverage by examining a number of small (but artificial) tasks
- danger that analysts do more of a user test on themselves than a CW
- does not find as many problems as user testing

Related methods

- Cognitive Jogthrough (a faster CW)
- Programming Walkthrough
- GOMS
- Heuristic Evaluation
- User Testing
- Thinking Aloud
Conclusions

• efficient evaluation method for early design
• do it quickly (otherwise not worthwhile)
• CW is no substitute for user testing
• use CW as prelude to user testing (eliminate obvious problems)