Human Computer Interaction

Evaluation Techniques

Lecture # 10

Imran Siddiqi
imran.siddiqi@gmail.com

Evaluation

- Tests usability and functionality of system
- Occurs in laboratory, field and/or in collaboration with users
- Evaluates both design and implementation
- Should be considered at all stages in the design life cycle

Evaluation

- Should occur throughout the design life cycle
- Evaluation results – Feedback for design changes
- Evaluation by
  - Designer/Usability Expert: early designs and prototypes
  - Actual users: Working prototype or implementation
- User may be involved in early designs as well

Goals of Evaluation

1. Assess extent/accessibility of system functionality
   - Functionality must accord with user requirements
   - Design – enable users perform the intended tasks more easily
   - Functionality – Available and clearly reachable
2. Assess user's experience of Interaction
   - How easy the system is to learn
   - Usability and user satisfaction
   - Enjoyment

Goals of Evaluation

3. Identify any problems
   - Unexpected results
   - Confusion amongst users
   - Related to both functionality and Usability

Evaluation Techniques

- Evaluation through
  - Expert Analysis
  - User Participation

Evaluation through Expert Analysis

- First evaluation – Before any implementation (Ideally)
- The later an error is discovered – the more costly it is
- Evaluation methods – Designers/experts
  - Identify areas that are likely to cause difficulties

Evaluation through Expert Analysis

- Cognitive Walkthroughs
- Heuristic Evaluations
- Model-based Evaluations
- Using previous studies in evaluation

Cognitive Walkthroughs

- Code walkthrough
  - Source Code inspection by reviewers
  - Following paths through the algorithms or code as determined by input conditions and choices made along the way
- Cognitive Walkthrough
  - Evaluator ‘steps through’ an action sequence to check an interface for potential usability problems
  - Steps: Sequence of actions to be performed in order to perform some task

Cognitive Walkthroughs

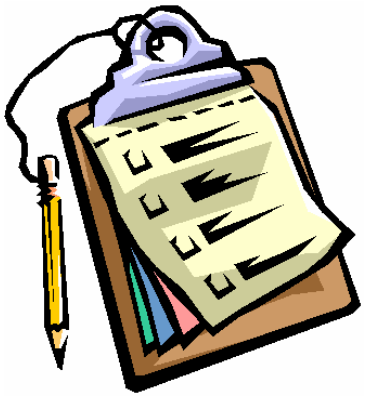
- Main focus – How easy the system is to learn
  - Learning through exploration

Cognitive Walkthrough - Inputs

1. Interface

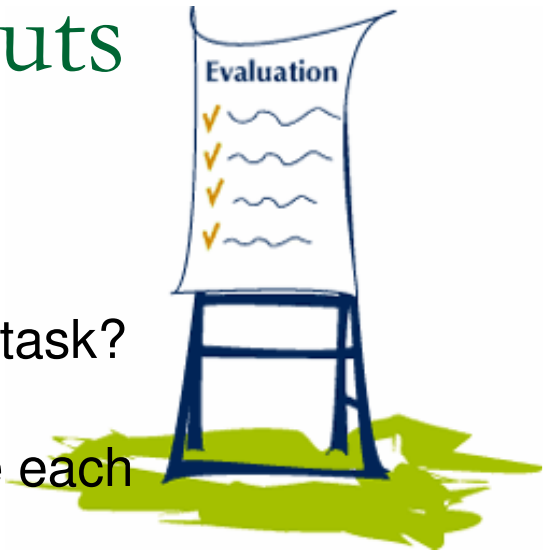
- What is to be evaluated?
- Implementation or Prototype
  - May not be complete but should be fairly detailed (e.g. location & wording of a menu)

2. Task(s)



- Description of the task(s) to be analyzed
- Should be representative task(s)

Cognitive Walkthrough - Inputs



3. Action Sequence

- What is the correct action sequence for each task?
- A complete list of actions needed to complete each task
- Example actions are: *"press the RETURN key"*, *"move cursor to 'File' menu"*
- Could also be a sequence of several simple actions that a typical user could execute as a block, such as, *"Select 'Save' from 'File' menu"*

Cognitive Walkthrough - Inputs

4. Users

- Who will be the users of the System?
- Background experience or technical knowledge of intended users
- Example user: *“Windows users who have worked with MS Word"*

Cognitive Walkthrough - Procedure

- Evaluator steps through the action sequence (‘3’)
- For each step, the evaluator tries to answer the following four questions:

Cognitive Walkthrough - Procedure

1. Is the effect of the action the same as the user's goal at that point?

- User performs some action – there is some effect
- Is it what the user was trying to do
- For example:
  - Action: A Button Pressed
  - Effect: Document Saved
  - Is it 'Saving the document' that the user wanted?

Cognitive Walkthrough - Procedure

2. Will users see that the action is available?

- Will users see the button/icon/menu item that is used to produce the action
- Only visibility is concerned – Not Recognition
- For example, Remote control – Buttons hidden under a covered panel

Cognitive Walkthrough - Procedure

3. Once the users have found the correct action, will they know it is the one they need?

- Complements the previous question
- Will the user recognize the button/icon

Cognitive Walkthrough - Procedure

4. After the action is taken, will users understand the feedback they get?

- User performs a correct action – will he know he has done so?
- Confirmation of the action performed
- Similar to the execution – evaluation cycle

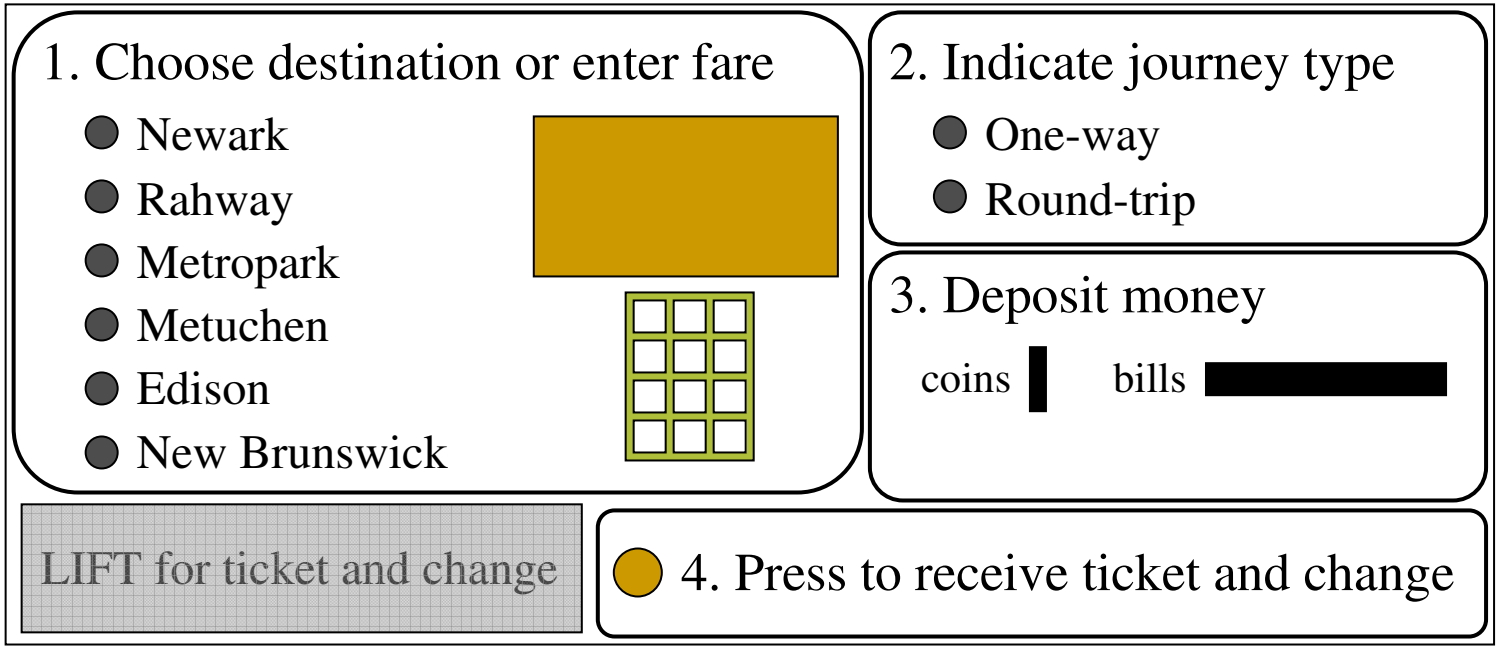
Cognitive Walkthrough - Documentation

- Prepare standard evaluation forms
- List the inputs (4 items in the first list), date & time of walkthrough, names of evaluators etc.
- For each action (item 3 in the list) prepare a separate form
  - Answers for the 4 questions in the second list
  - Negative answers – document separately on the problem sheet
  - Indicate the severity of the problem – how likely/serious it is

Cognitive Walkthrough – Example 1
Ticketing Machine

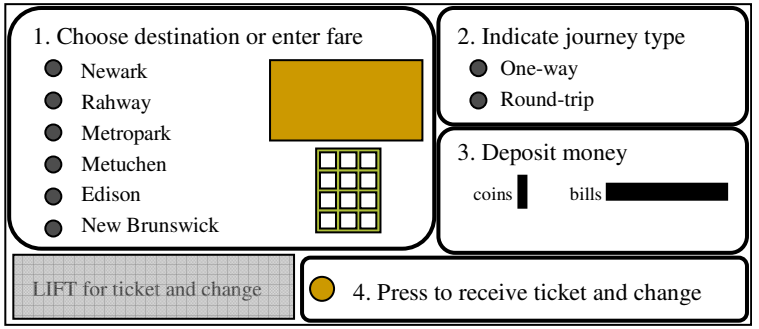
Cognitive Walkthrough - Example

- Consider the following interface for a train ticket vending machine



Cognitive Walkthrough - Example

- Scenario
  - Nick is visiting a friend for a few days and wants to buy a one-way ticket to *New Brunswick.* Nick has a $5 bill in his pocket plus some loose change.



Cognitive Walkthrough - Example

- Interface
  - Standard size ticketing machine with push buttons, numeric keypad, LCD Display ….
- Task
  - Purchase a one-way ticket to New Brunswick
- Action Sequence
  - Press the button next to New Brunswick to chose the destination
  - ….

Cognitive Walkthrough - Example

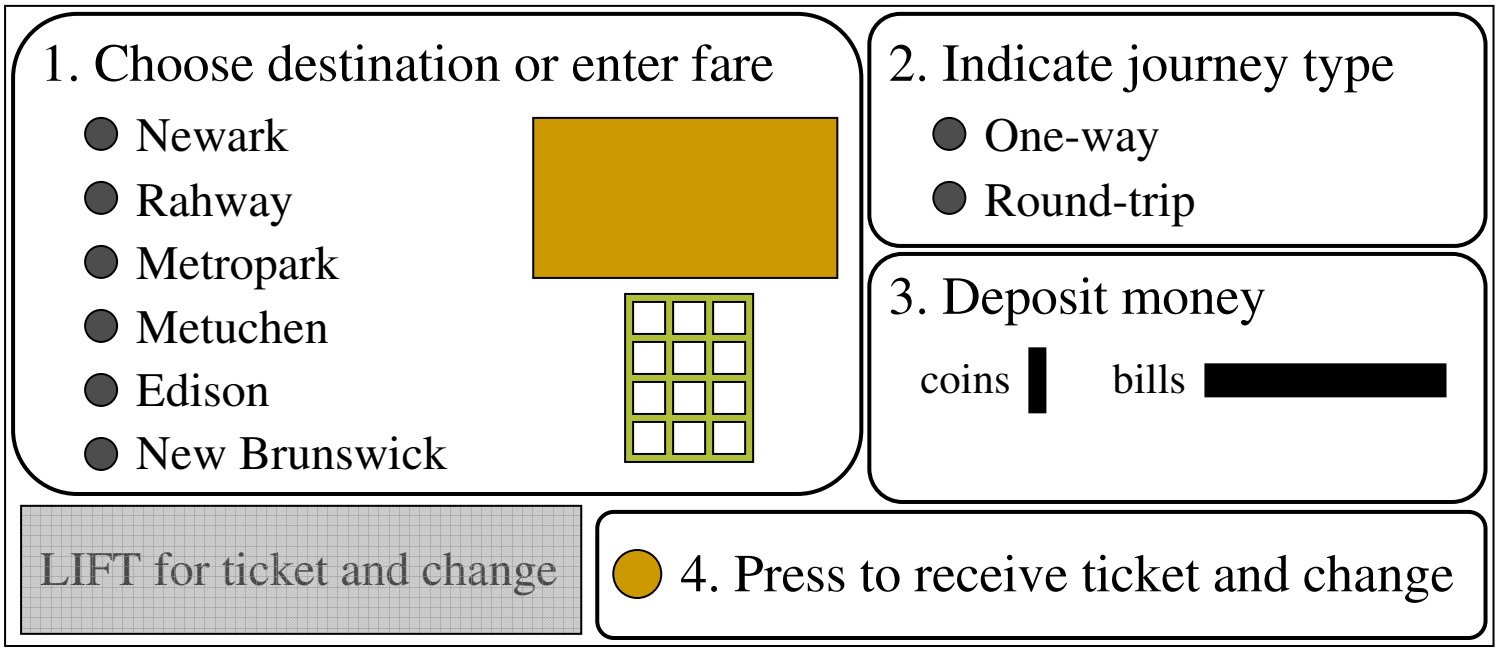
- Users
  - Ages 12+
  - Normal vision
  - Understanding of English

Cognitive Walkthrough - Example

- Action: Press the button next to New Brunswick to chose the destination
- Answer the four questions for this action

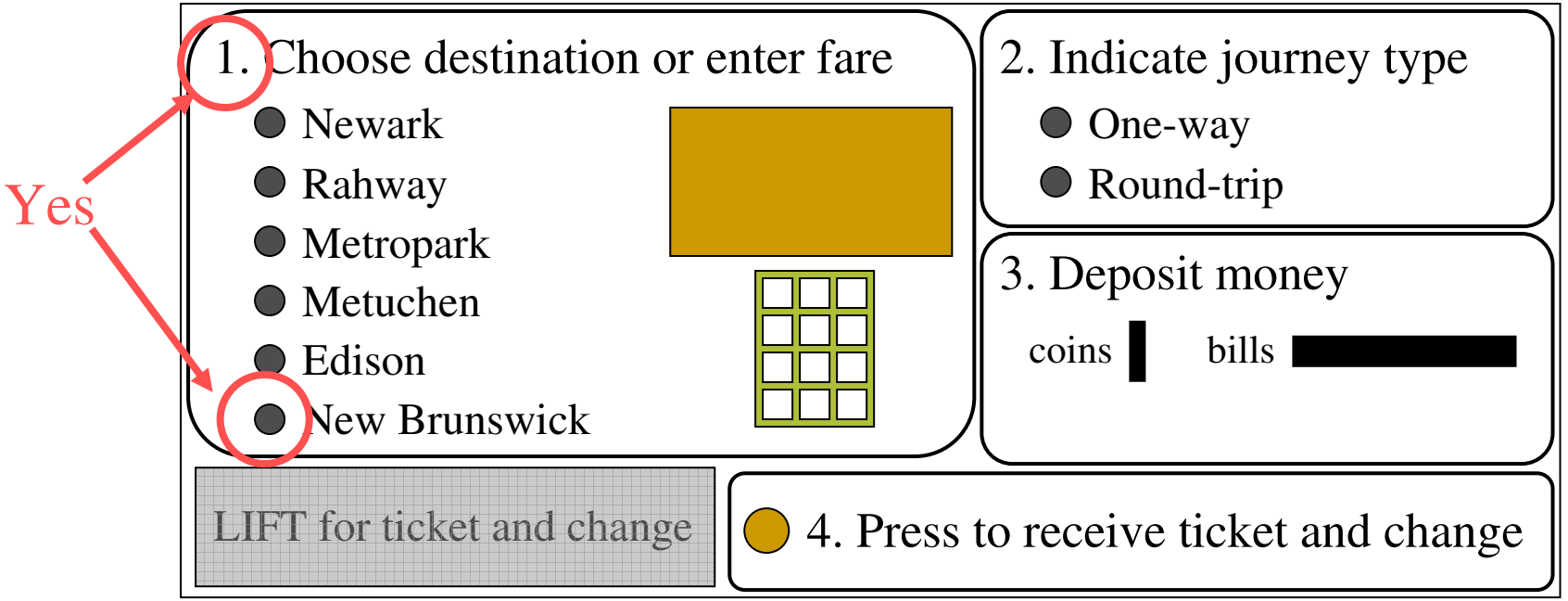
1. Is the effect of the action the same as the user's goal at that point?

YES



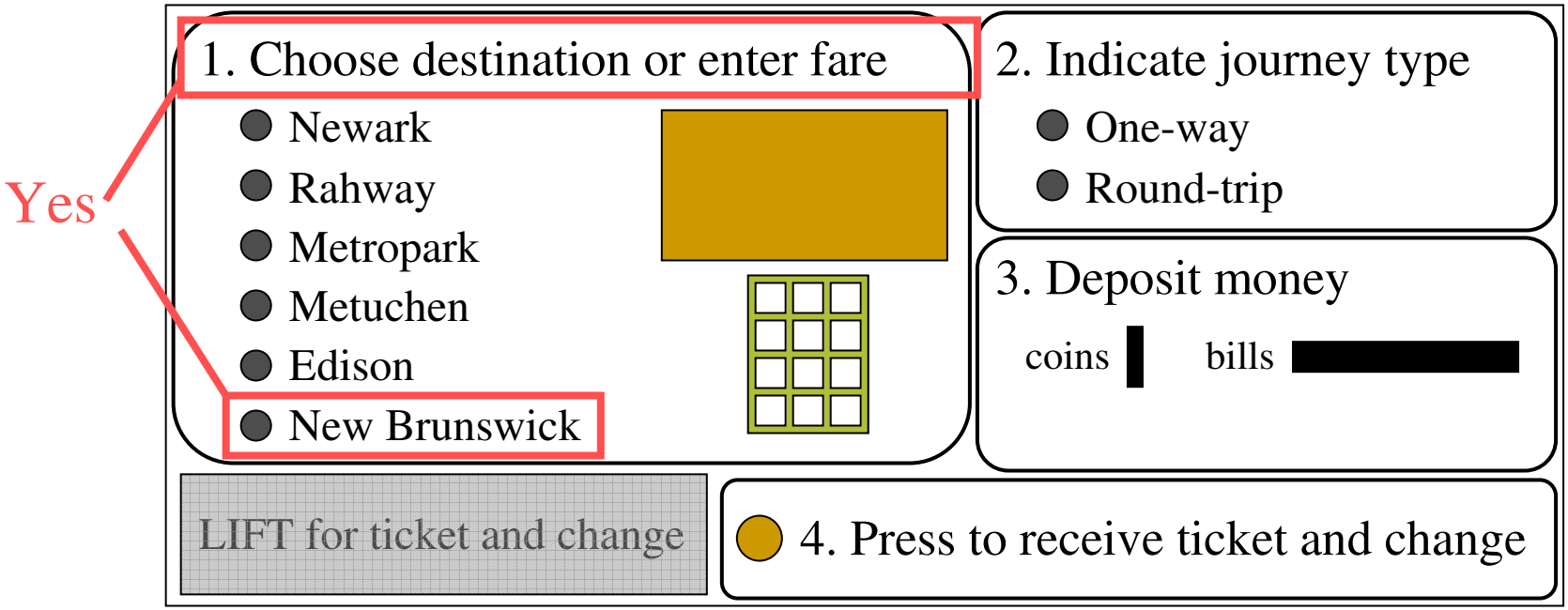
Cognitive Walkthrough - Example

2. Will users see that the action is available?



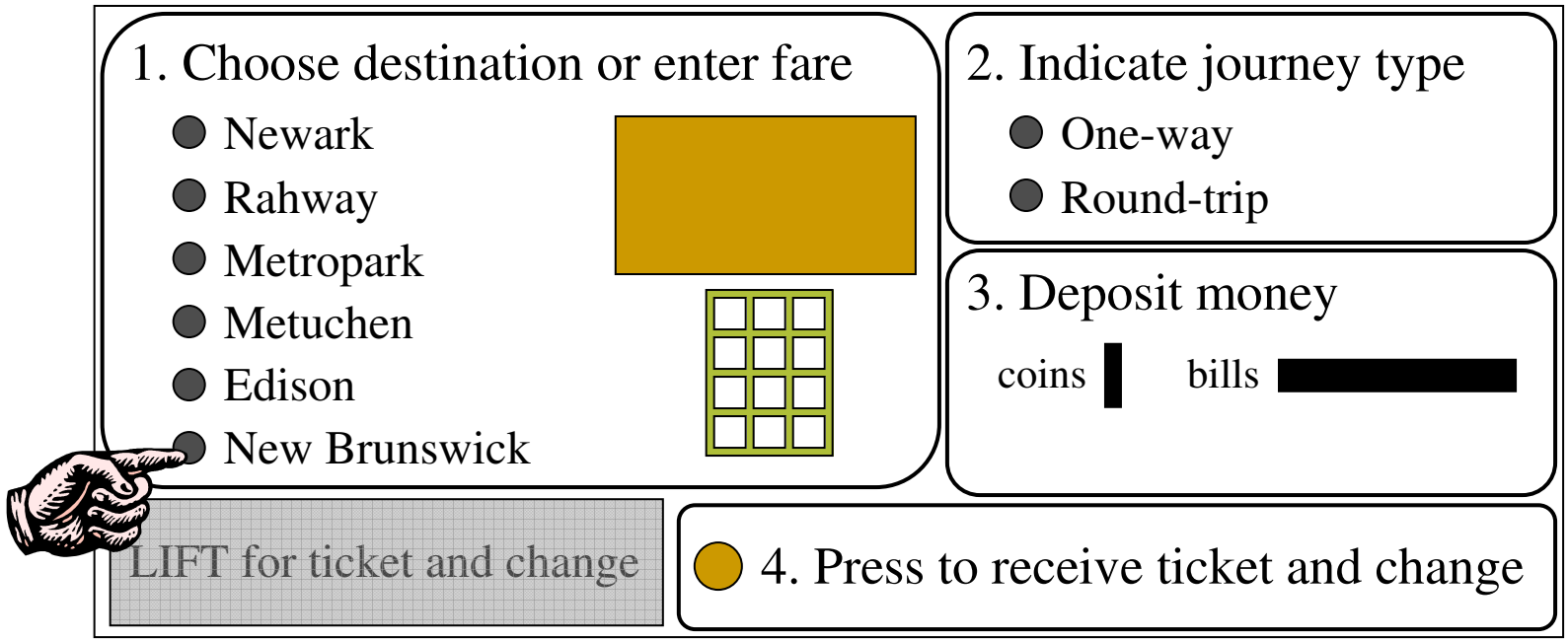
Cognitive Walkthrough - Example

3. Will users recognize what they have located?



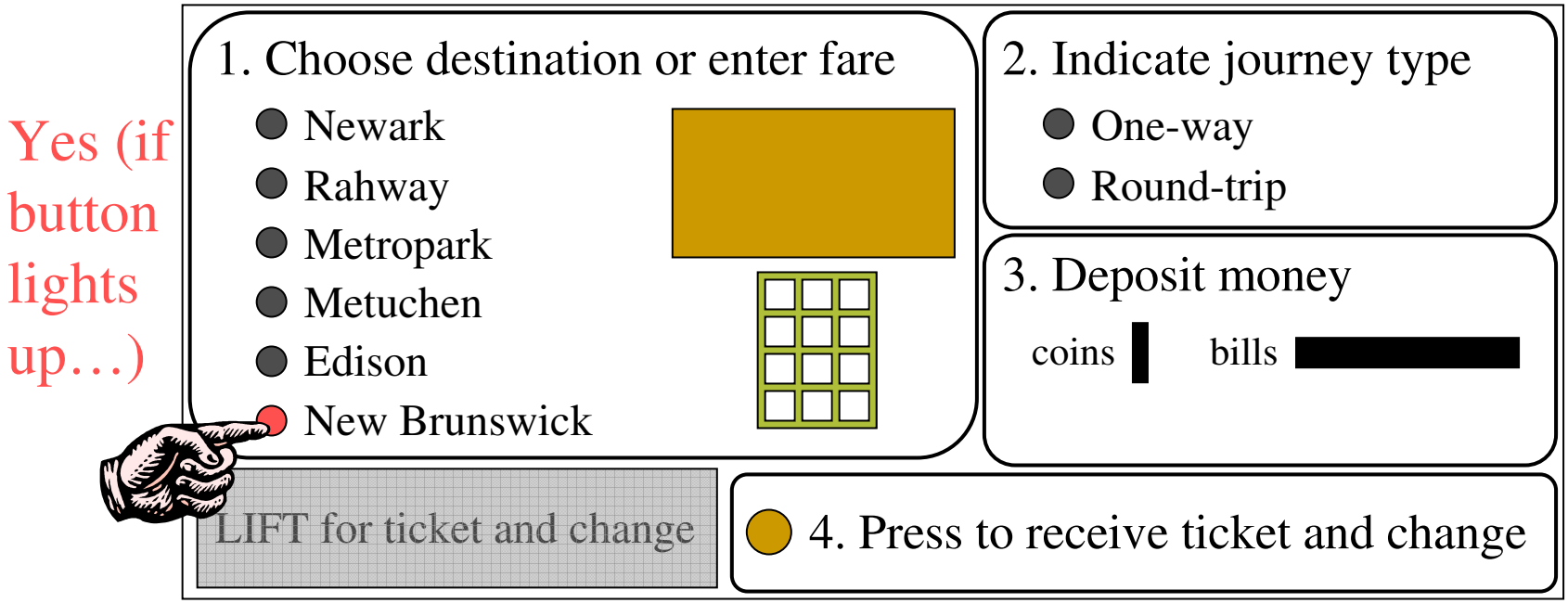
Cognitive Walkthrough - Example

4. Will users understand the feedback after the action?



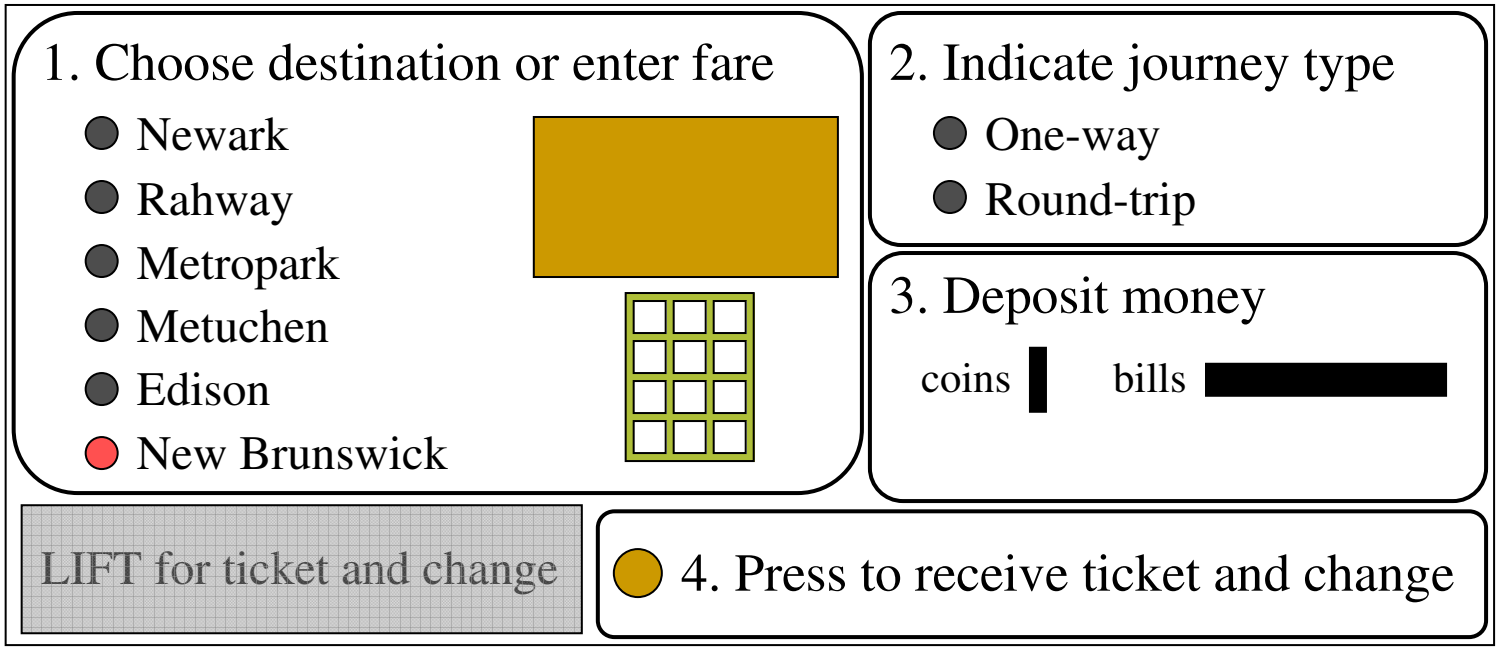
Cognitive Walkthrough - Example

4. Will users understand the feedback after the action?

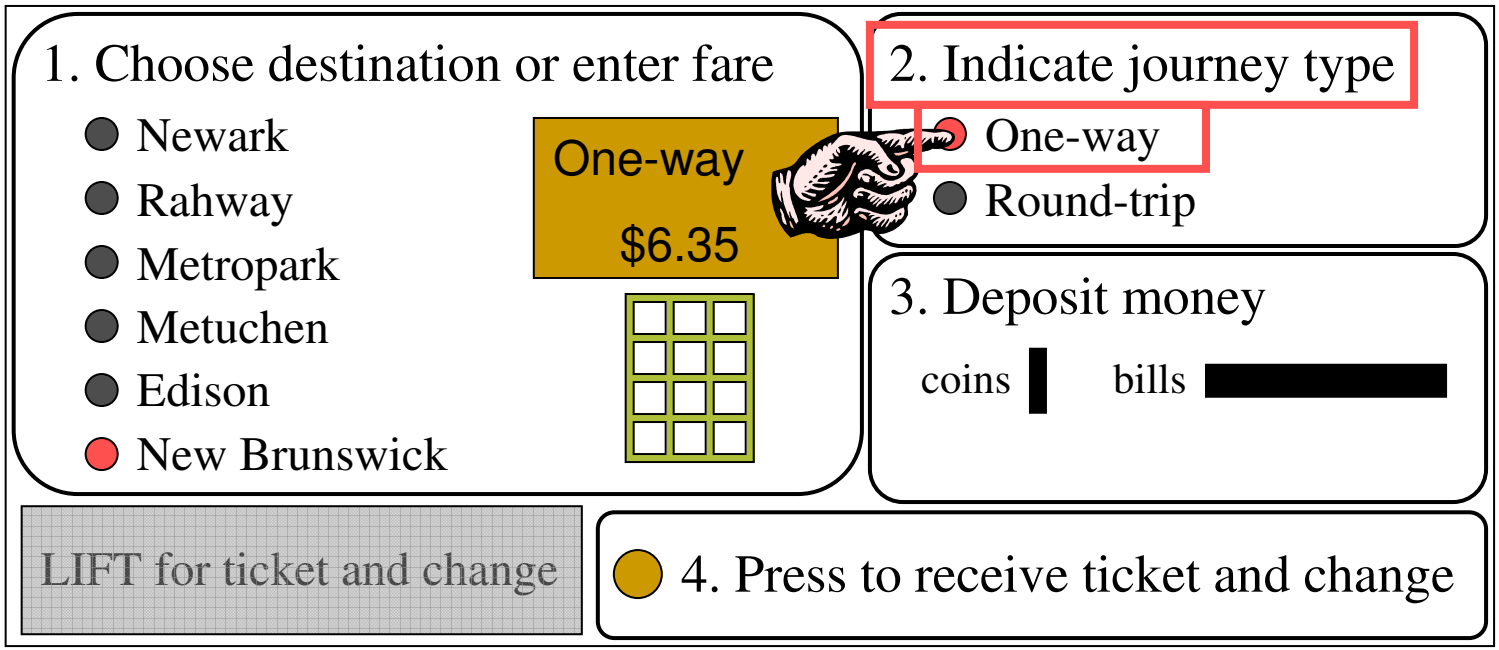


Cognitive Walkthrough - Example

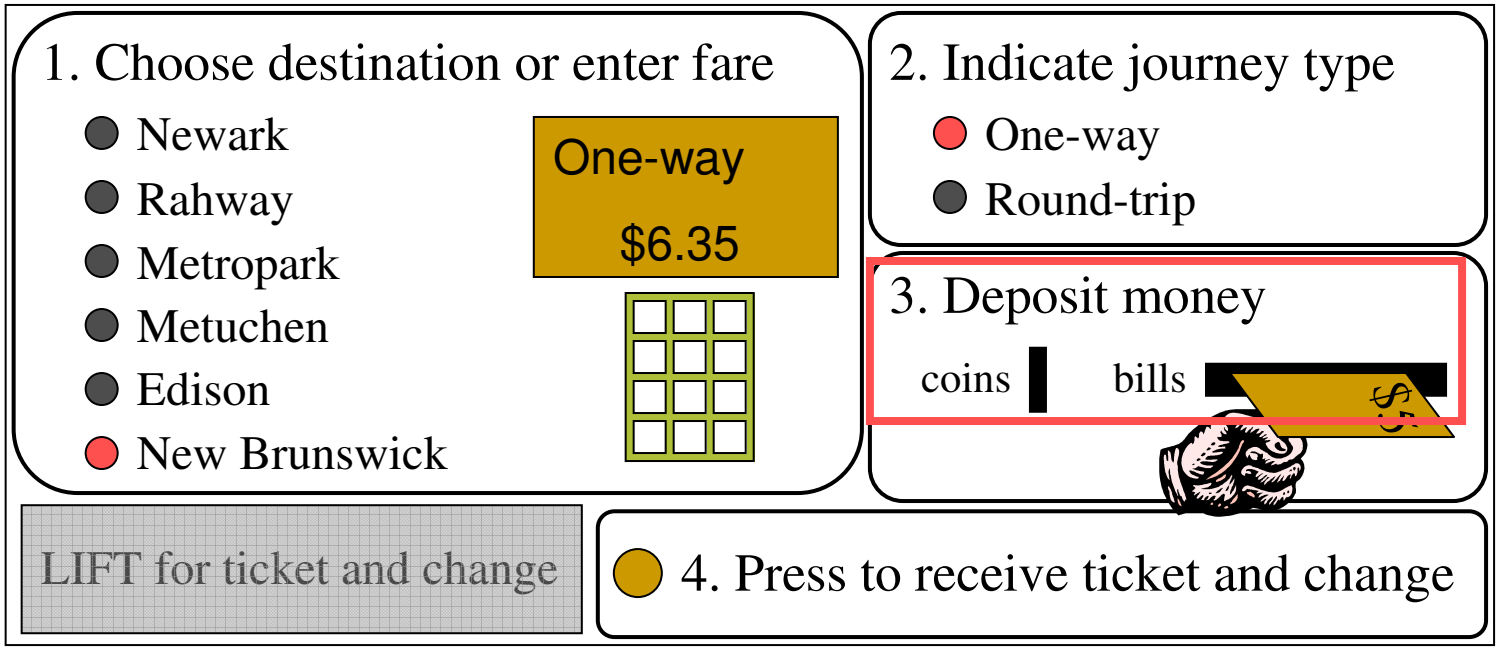
- Perform the same for each of the steps in the action sequence



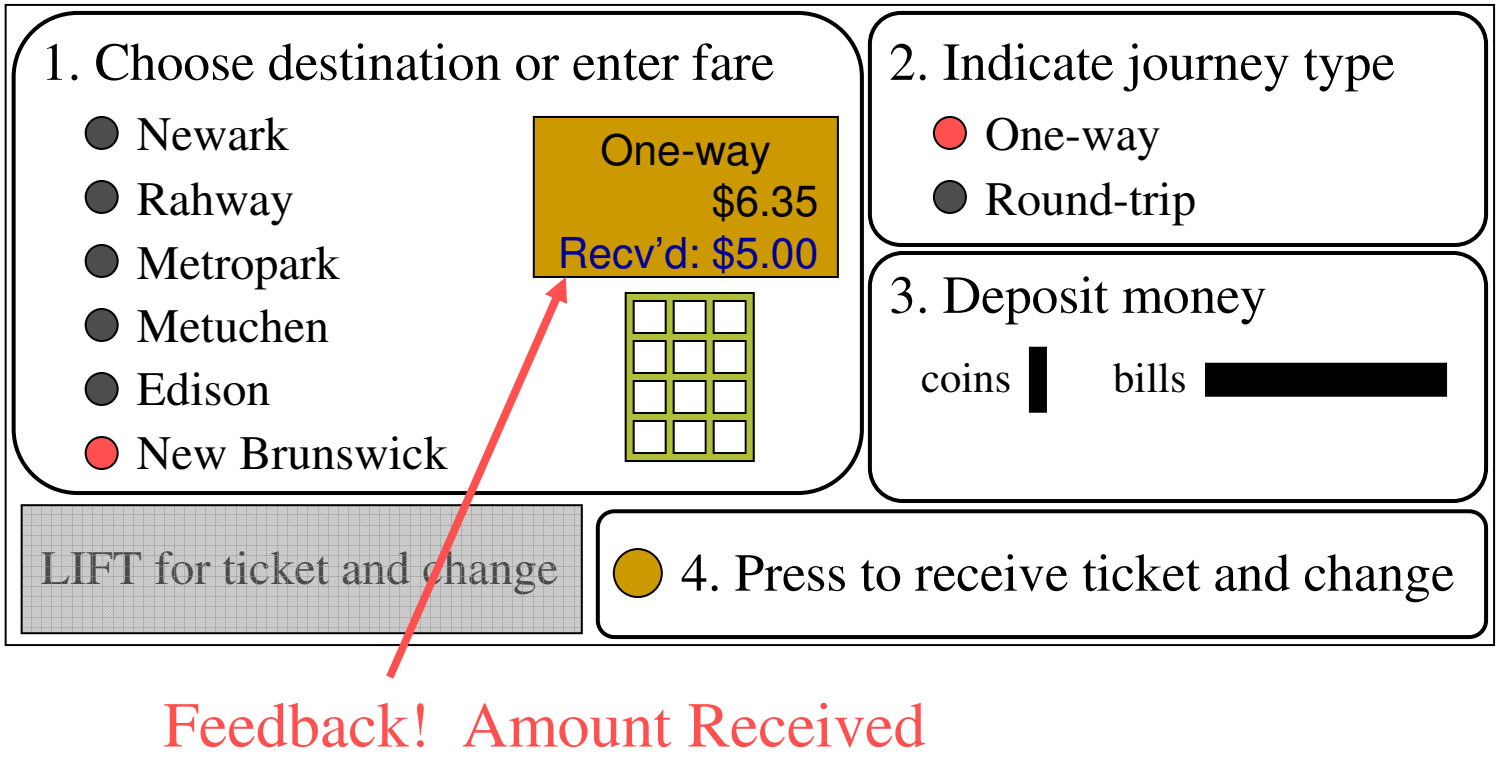
Cognitive Walkthrough - Example



Cognitive Walkthrough - Example



Cognitive Walkthrough - Example



Cognitive Walkthrough - Example

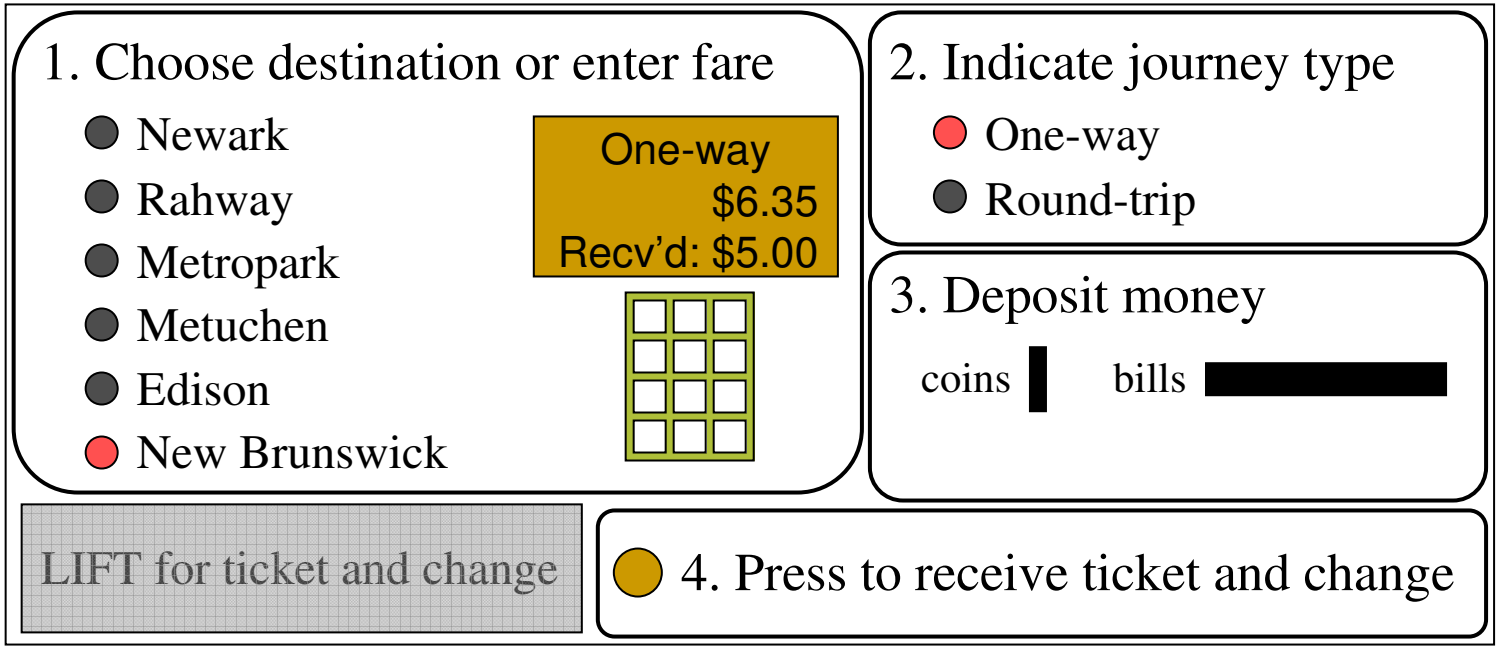
- Finish with a normal case first (e.g. completion of purchase)
- Also be sure to handle common error cases
- Examples:
  - What if Nick realized he didn't have enough money before putting it in?
  - What if Nick realizes he doesn't have enough only after putting in money?

Cognitive Walkthrough - Example

Not enough $…

What does the user want to achieve?

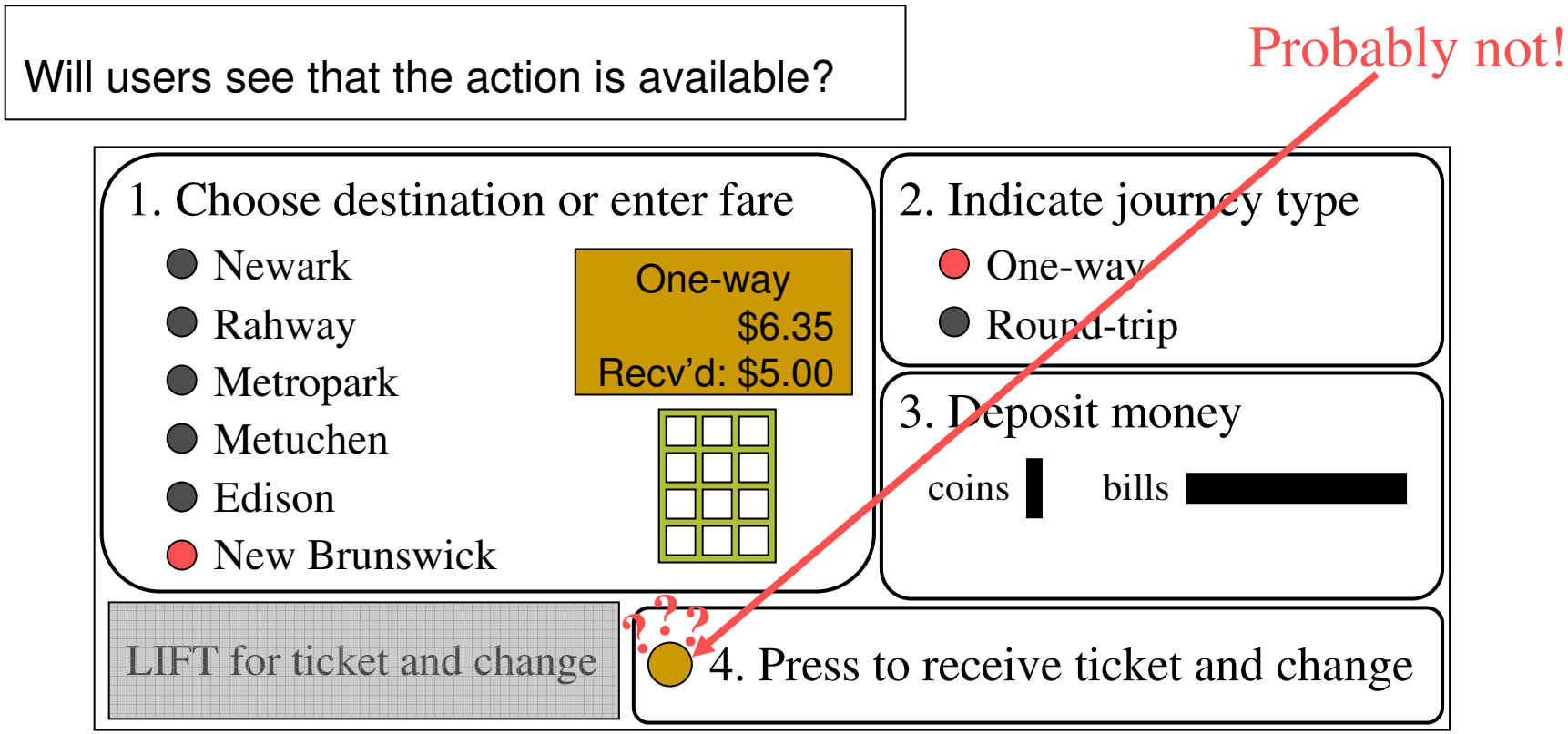
Cancel and get money back



Cognitive Walkthrough - Example

Not enough $…

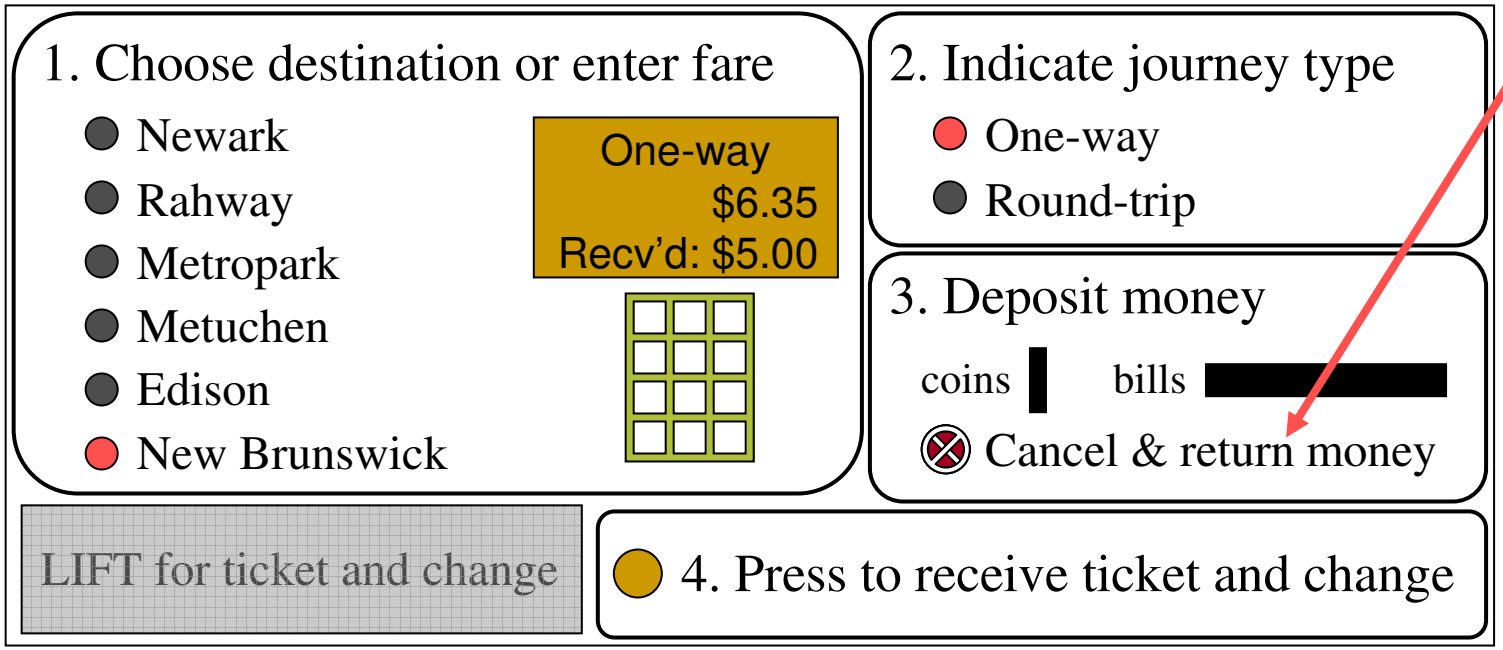
Cancel and get back money



Cognitive Walkthrough - Example

Will users see that the action is available?

Add a new UI element to make it clear!



Cognitive Walkthrough – Example 2
Call Forwarding

Forwarding calls on a campus telephone system, from the perspective of a first time user [Wharton et al., 1994]

Cognitive Walkthrough - Example

- Interface
  - Standard-size, touch-tone phone on desk. Overlay template includes the following information:
    - FWD *2
    - CNCL #2
    - SEND ALL *3
- Task
  - Cancel current forwarding and forward calls instead to a colleague with the extension 1234.

Cognitive Walkthrough - Example

- Action Sequence

1. Pick up the receiver.

Phone: *dial tone*

2. Press #2. Command to cancel forwarding

Phone: *bip bip bip*

3. Hang up the receiver

4. Pick up the receiver

Phone: *dial tone*

5. Press *2. Command to forward calls

Phone: *dial tone*

6. Press 1234.

Phone: *bip bip bip*

7. Hang up the receiver.

Cognitive Walkthrough - Example

- Users
  - Old/New faculty, staff, guests, and visitors
  - For this evaluation assume that the user is a new university professor

Cognitive Walkthrough - Example

- Pick up the receiver
- Phone: *dial tone*

- Answer the four questions for this action

Seems **OK** based on prior experience with phones.

Cognitive Walkthrough - Example

- *Press #2.*
- *Phone: bip bip bip*
- Answer the four questions for this action

1. Is the effect of the action the same as the user's goal at that point?

… But how does the user even know that forwarding is in effect?

2. Will users see that the action is available?

Probably yes, CNCL is visible on the template.

Cognitive Walkthrough - Example

- *Press #2.*
- *Phone: bip bip bip*

3. Will users recognize what they have located?

•Might not recognize CNCL as the control to cancel forwarding
•Might think that just pressing '2' is sufficient, instead of '#2'
•Might try to press the buttons simultaneously, rather than sequentially

4. Will users understand the feedback after the action?

How do first-time users know they have succeeded? After some experience, they will recognize the *bips* as confirmation, but will they at first?

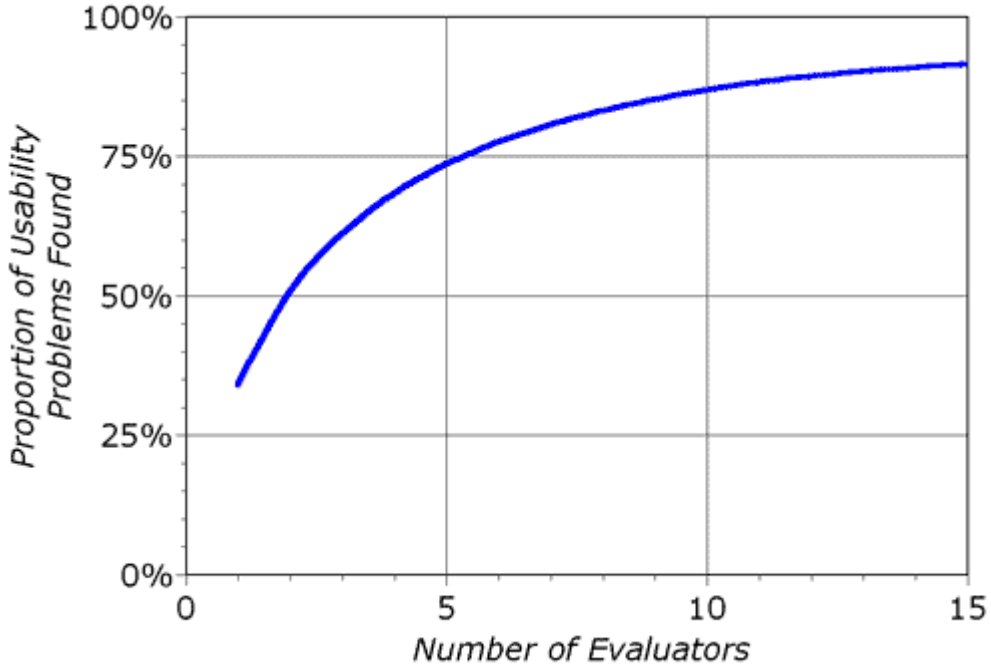
Heuristic Evaluation

Heuristic Evaluation

- Heuristics
  - Guidelines, General Principle, Rule of thumbs
- Heuristic Evaluation
  - Critique of a system using a set of simple and general heuristics
  - Developed by Neilsen and Molich

Heuristic Evaluation

- Several evaluators independently critique a system for potential problems
- Neilsen's Experience: 3-5 evaluators



Heuristic Evaluation

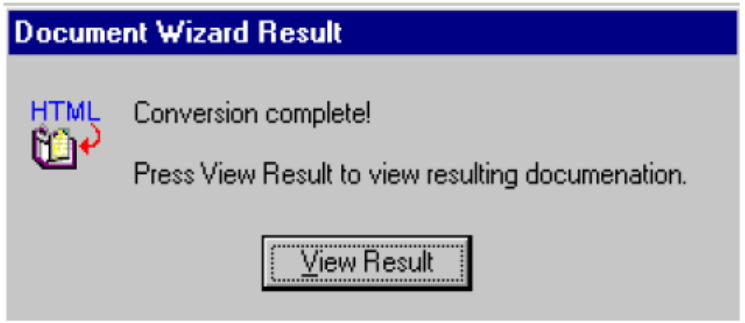
- Neilson's 10 heuristics
- Can be supplemented by domain specific heuristics if required
- Evaluators
  - Asses the system and note the violations of these heuristics
  - Also asses the severity of the problem

Heuristic Evaluation

- Problem Severity - Rating on a scale 0-4
  - 0 = I do not agree that this is a usability problem
  - 1 = Cosmetic Problem only, need not to be fixed unless extra time is available
  - 2 = Minor Usability Problem, Low priority fixation
  - 3 = Major Usability Problem, High priority fixation
  - 4 = Usability Catastrophe, Imperative to fix

Nielsen's 10 Heuristics

- 1. Visibility of System Status
  - Always keep user informed about what is going on?
  - Appropriate feedback – How long/ how much completed
  - Cursor Change, Progress Bar
  - Do not OVERDO it!



Nielsen's 10 Heuristics

- 1. Visibility of System Status (Contd…)
  - Response Time
    - < 0.1 secs – Instantaneous
    - 0.1 – 1 secs – User notices, but no feedback needed
    - 1 – 5 secs – Display busy cursor
    - > 1 - 5 secs – Display progress bar

Nielsen's 10 Heuristics

- 2. Match between system and the real world
  - *System should speak user's language not system oriented terms*
  - *Words, phrases and concepts – familiar to the user*

Nielsen's 10 Heuristics

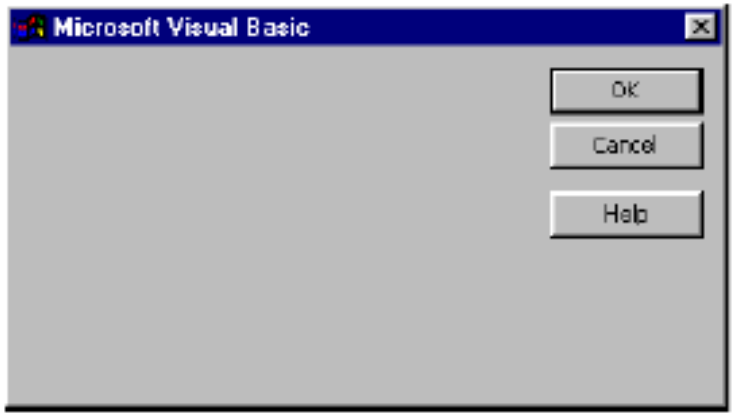
- 3. User control and freedom
  - *Users often make mistake and will need a clearly marked "emergency exit" to leave the unwanted state*
  - *Support undo and redo*
  - *Long operations should be cancelable*
  - *All dialogs should have a cancel button*

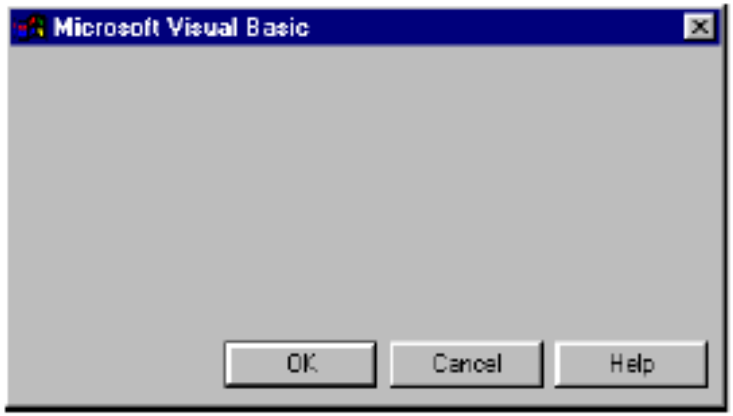
Nielsen's 10 Heuristics

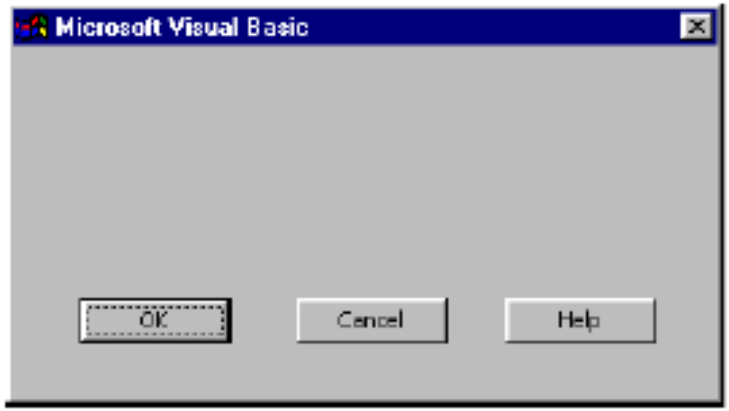
- 4. Consistency and Standards
  - *Principle of least surprise*
  - *Similar things should look and act similar*
  - *Different things should look different*
  - *Properties*
    - *Terminologies, Size, Location, Color, …*
  - *Follow standards*

Nielsen's 10 Heuristics

- 4. Consistency and Standards







Nielsen's 10 Heuristics

- 5. Error Prevention
  - *Make it difficult to make errors*
  - *Better than good error messages is careful design which prevents a problem from occurring in the first place*
  - Selection is less error-prone than typing
    - Don't overdo it

Nielsen's 10 Heuristics

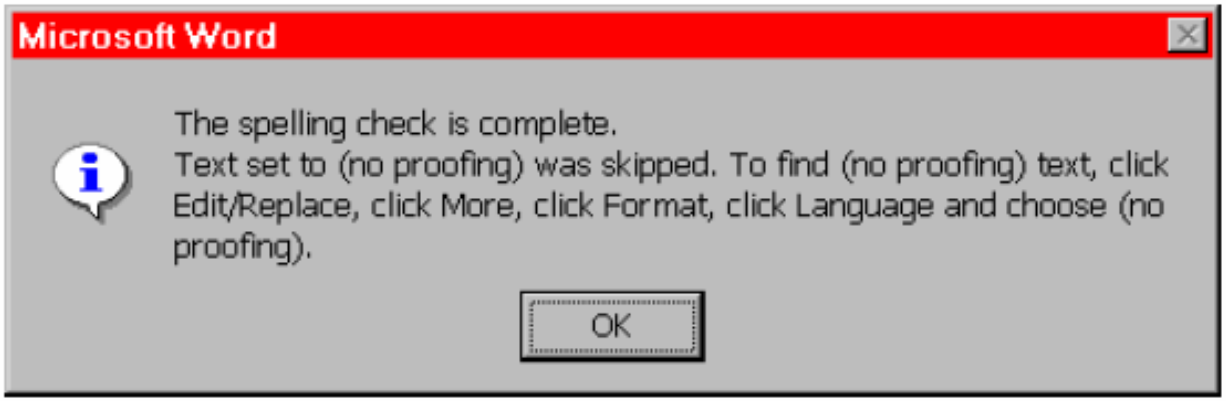
- 5. Error Prevention (Contd…)
  - *Disable illegal commands*
    - Copy is impossible if nothing is selected – then the command should be disabled ("grayed out") so that it simply can't be selected in the first place.
  - *Avoid Modes*
    - If cant avoid – make it visible

Nielsen's 10 Heuristics

- 6. Recognition rather than recall
  - *Minimize the user's memory load by making objects, actions, and options visible*
  - *Instructions for use of the system should be visible or easily retrievable whenever appropriate*
  - *Use menus not command language*
  - *Use combo boxes not text boxes*

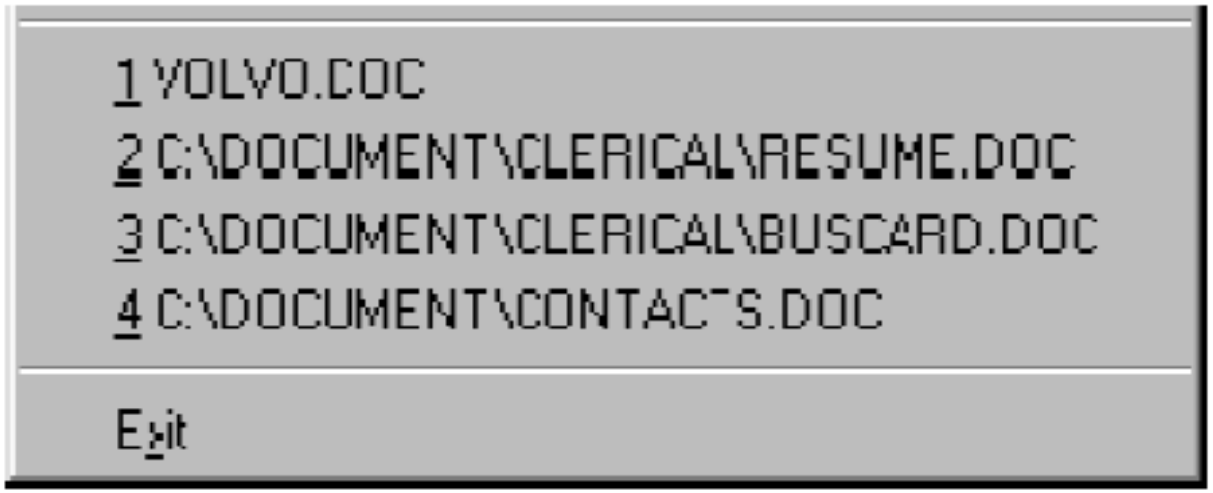
Nielsen's 10 Heuristics

- 6. Recognition rather than recall
  - *Over reliance on user's memory*
  - *Modal dialog box, so the user can't start following its instructions until after clicking OK*
  - *Clicking OK the instructions vanish from the screen, and the user is left to struggle to remember them*



Nielsen's 10 Heuristics

- 7. Flexibility and Efficiency
  - *Accelerators to speed up the interaction for expert user*
  - *Can cater to both inexperienced and experienced users*
    - Keyboard accelerators, Command abbreviations, Bookmark, History



Nielsen's 10 Heuristics

- 8. Aesthetic and minimalist design
  - *Less is More*
  - *Do not use information which is irrelevant or rarely needed*
  - *Every extra unit of info competes with the relevant units of info – diminishes their relative visibility*
  - *Omit extraneous info, graphics, features*

Nielsen's 10 Heuristics

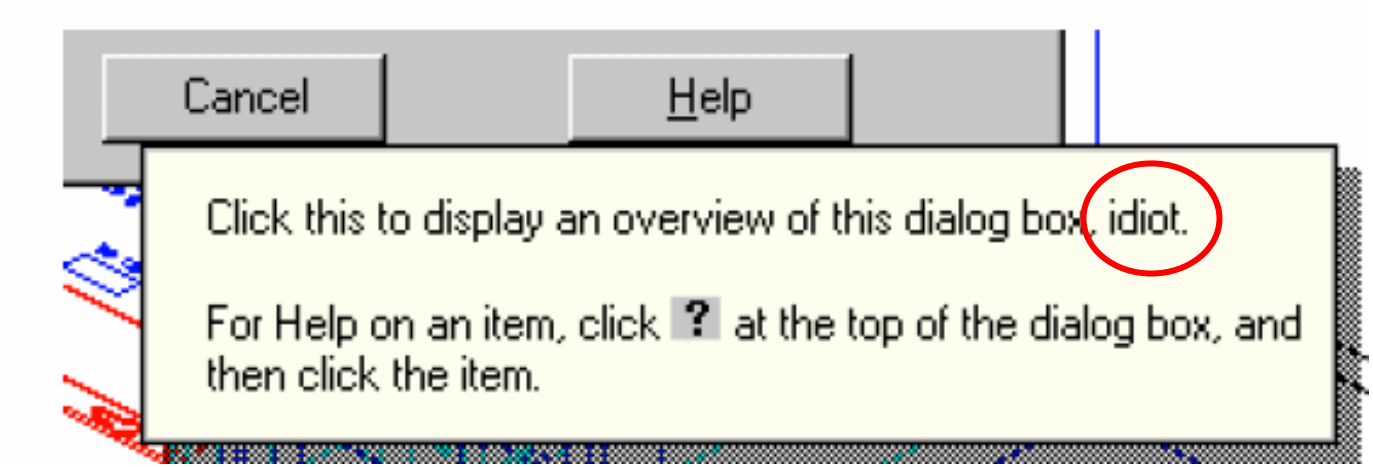
- 8. Aesthetic and minimalist design

Nielsen's 10 Heuristics

- 9. Help users recognize, diagnose, and recover from errors
  - *If you cant prevent errors – give a good error message*
  - *Should be in plain language (No codes)*
  - *Be precise – Restate user's input*
    - Not "Cannot open file", but "Cannot open file named paper.doc"
  - *Give constructive help*
    - Why error occurred and how to fix it
  - Be polite and non-blaming
    - Not "fatal error", not "illegal"

Nielsen's 10 Heuristics

- 9. Help users recognize, diagnose, and recover from errors (Contd …)
  - The tooltip shown comes from a production version of AutoCad!
  - Inserted by a programmer as a joke, but somehow never removed before release.



Nielsen's 10 Heuristics

- 10. Help and Documentation
  - *Users don't read manuals*
    - Prefer to spend time working toward their task goals, not learning about your system
  - *But manuals and online help are vital*
    - Usually when user is frustrated or in crisis
  - *Help should be:*
    - Searchable, Concrete and Short

References

- Chapter 9 - Human Computer Interaction by Dix et al.
- Usability Engineering – Rob Miller, MIT
- HCI & Interface Design – Jacek Gwidza