Computer Graphics and Human Computer Interaction

Lecture 4 Usability Methods

Manasawee Kaenampornpan

manasaweek@gmail.com

http://www.italpha.msu.ac.th/manasawee

About this course

- Purpose

    This subject examines the design, evaluation and implementation of interactive computing systems for human use (HCI) and the major phenomena surrounding them. Also considered are joint performance of tasks by humans and machines, structure of human machine communication, social and organizational interactions with machine design, human capabilities to use machines including their learn ability, engineering concerns that arise in designing interfaces, the process of specification design, implementation and evaluation of interfaces and design tradeoffs.

- Assessment

    - Individual courseworks (40%)

    - Quiz (10%)

    - Final (50%)

About this course

- Topics
  - Introduction
  - Frameworks for Cognition and Theories
  - Usability
  - Graphics and Sound
  - Design Methods and Process
  - Usability testing
  - CSCW
  - Mobile and Ubiquitous Interaction

Usability Evaluation

Usability evaluations can be conducted at many stages during and after the design and development process. In choosing a method, it is important to calculate the cost not only in terms of time and materials involved, but also in terms of the impact on the end-users, especially considering the cost of losing

return visitors to your website due to unusable design.

Usability Methods

- There are a variety of approaches to usability evaluation that you may choose to take. The methodologies can be divided into two broad categories: those that gather data from actual users and those that can be applied without actual users present.

Your choice of method depends on:

  - Cost of evaluation
  - Appropriateness to project
  - Time constraints
  - Cost of implementation
  - Cost of training new users

Usability Methods

- Personas
- Scenarios
- Task analysis
- Cognitive walkthrough
- User Interface Models
- Prototyping
- Card Sorting
- Focus group

Personas

- Composite archetypes based on behavioral data gathered from many actual users through ethnographic interviews

- Hypothesize personas from preliminary data as the basis for the initial interviewing process; refine as new information becomes available

- Personas + Goals = Models of Archetypal users and their purposes

- Goals are the "what" and Tasks are the "how"

- Scenarios are used to step the personas through tasks in order to achieve goals

Personas

- What different sorts of people might use this product?
- How might their needs and behaviours vary?
- What ranges of behaviour and types of environments need to be explored?
- Roles in business and consumer domains
- Behavioural and demographic variables
- Domain expertise vs. technical expertise
- Environmental considerations

Research paper: Experimenting with the use of persona in a focus group discussion with older adults in Malaysia http://dl.acm.org/citation.cfm?doid=1738826.1738889

Kivio Users

	The researcher	The Sysadmin	The OSS developer	The CS student
Name	Alexander Weiß	Donald M. Berry	Kristian Larsson	Eric Neville
Age	30	30	26	24
Location	Germany	US	Sweden	France
Social Life	Alexander lives with his girl-friend in a flat in Hamburg.	Donald lives with his wife and 1-year old daughter in a house in Portland.	Kristian shares an apartment with two friends in Stockholm. His girl-friend lives in Uppsala. They see each other every weekend.	Eric lives with his parents in a small city close to Lyon. He visits the university there. Often, he stays at his friend's apartment for playing PC games and programming.
Work Life	He works at centre for environmental systems research and designs plans for replacable energies in a EU-funded project.	He is a lead system administrator in a huge network solutions company in Portland.	A software developer with a dayjob in a medium-sized software company. Works on KDE in his spare time.	He is a student of computer science. Besides university, he performs small programming jobs for people in his neighbourhood.
Computer Experience	All are highly experienced with computers.			
Time at a computer per week	26-50 hours per week	35-50++ hours per week	30-50++ hours per week	25-45 hours per week
Computer tasks	Office tasks and Field-dependent. Also educational and recreational. No development.	Development and network administration. Does not use PC for office tasks, educational, and even recreational.	Mostly development and recreational. Also network administration and office.	Mostly development. Also educational, recreational, and network administration. Does not use for office work.
Relation to OSS.	He is not passionate about OSS.	He is a convinced user of OSS.	He is involved with OSS development.	He is a convinced user of OSS.
Requirements wrt diagramming	office requirements	highest claims	easy-going	eager beaver
Frequency of drawing diagrams	Each 2nd month	Twice per month	Once a month	Each 2nd month
Diagram main type	Flowcharts. Also visualising thoughts. No technical ones.	All, except sitemap	Visualising thoughts	Diagrams mostly UML.
Size and complexity	15-20 elements, 2-3 levels, 3-7 shapes	15-30 elements, 2-5 levels, 4-9 shapes	15-20 elements, 2-3 levels, 3-7 shapes	15-20 elements, 2-3 levels, 3-7 shapes
Diagram purpose and context.	For non-IT job.	Diagrams are for the IT-Job, never for himself.	Diagrams are mostly for himself but, also for formally presenting.	For formally presenting in university. Not for himself, as work input or for any jobs
Current diagramming	Power Point or OOo.	Visio.	Pen and Paper.	Dia, Umbrello.

Goals

- **Corporate goals**
  - Increase profit and market share
  - Defeat competition
  - Hire more people
  - Offer more products
- **Personal goals**
  - Not feel stupid
  - Not make mistakes
  - Get an adequate amount of work done
  - Have fun (or at least **Practical gonaolst be too bored)**
- **(bridge gap between personal and corporate goals)**
  - Avoid meetings
  - Handle the client's demands

Scenarios

"A scenario is a concise description of a persona using a product to achieve a goal" (Cooper).

- *concise* : short but complete; breadth instead of depth

- *product* : assume the product (software or physical device) exists, even if it doesn't

- *goal* : the reason why we perform a task

Scenarios : Why Use Them?

- Scenarios help us validate our design

- Scenarios help us check our assumptions

- Successful Scenarios help us transfer theoretical/conceptual design to "wire frame" design

- Like Personas, Scenarios create a shared understanding of the end users –for the entire design team (including designers, marketing folk, engineers, executives, etc).

- Like Personas, Scenarios emphasize that The User Is Not Like Me.

Example

- http://www.uiaccess.com/accessucd/scenarios_eg.html

Scenario 13
Task: Change retirement account investments
Persona: Roger
User Group: Retiree
Date: October 2003

Background
HRWeb is a Human Resources (hr) Management web-based application used by Acme Insurance. HRWeb handles hr management such as employee records, compensation, and benefits management. User Groups for HRWeb include: hr managers, hr specialists, hr administrative assistants, non-hr managers, non-hr administrative assistants, employees, retired employees.

Scenario
Roger has been tracking the increase in natural gas prices on the Web. He decided that he'd like to exchange some shares out of the Blue Chip Fund in his retirement account into a mutual fund that specializes in natural gas. Having researched natural gas on the Web, Roger feels that the Specialty Natural Gas Fund is safe and this would be a great time to purchase the fund.

Roger is working on his home computer on Wednesday morning. He has about an hour before his tee time, so he decides to make the changes in his retirement account that he's been thinking about.

Roger opens Internet Explorer. He clicks Favorites and selects HRWeb.

Research papers on using scenario in design

- ➡ Scenario-Based Design by Mary Beth Rosson and John M. Carroll https://www.e-education.psu.edu/drupal6/sites/www.e-education.psu.edu.geog583/files/Rosson_2002_sbd.pdf

Task analysis

- Methods of analysing people's jobs:
    - what people do
    - what things they work with
    - what they must know

- *Watch VDO: A Task Analysis*

An Example

- in order to clean the house
  - get the vacuum cleaner out
  - fix the appropriate attachments
  - clean the rooms
  - when the dust bag gets full, empty it
  - put the vacuum cleaner and tools away

- must know about:
  - vacuum cleaners, their attachments, dust bags,
    cupboards, rooms etc.

Differences from other techniques

Systems analysis   **vs.**   **Task analysis**

system design   -   focus   -   the user

Cognitive models   **vs.**   **Task analysis**

internal mental state   -   focus   -   external actions

practiced `unit' task   -   focus   -   whole job

Task Analysis

- Find out
  - who users are
  - what tasks they need to perform
- Observe existing work practices
- Create scenarios of actual use
- Try-out new ideas before building software

Why Task Analysis?

- System will fail if not done?
  - does not do what the user needs
  - is inappropriate to the user
- "System must match users' tasks"
- Why not define "good" interfaces?
  - infinite variety of tasks and users
  - guidelines are usually too vague (e.g.,"give adequate feedback")

Questions

- Who is going to use system?

- What tasks do they now perform?

- What tasks are desired?

- How are the tasks learned?

- Where are the tasks performed?

- What's the relationship between user & data?

Questions (cont.)

- What other tools does the user have?
- How do users communicate with each other?
- How often are the tasks performed?
- What are the time constraints on the tasks?
- What happens when things go wrong?

Who?

- Identity?
  - in-house or specific customer is easy
  - need several typical users for broad product
- Background
- Skills
- Work habits and preferences
- Physical characteristics

Who (ATM Machine)?

- Identity?
    - people who have ATM card
- Background
    - have an bank account
- Skills
    - know how to put cards into ATM machines

Who (ATM cont.)?

- Work habits and preferences
  - not applicable
- Physical characteristics
  - varying heights –> don't make it too high or too low!

Talk to Them

- Find some real users
- Talk to them
    - find out what they do
    - how would your system fit in
- Are they too busy?
    - buy their time
        - t-shirts, coffee mugs, etc.

What Tasks?

- Important for both automation & new functionality

- Relative importance of tasks?

- Observe users

  - on-line billing example

    - small dentists office had billing automated

    - assistants were unhappy with new system

    - old forms contained hand-written margin notes

What Tasks (ATM machine)?

- Old tasks?
    - Put cash into account at bank
    - Take cash out from account at bank
- New tasks?
    - ATM card to
        - Put cash into account at bank
        - Take cash out from account at bank
        - new currency! Different languages
- Level of detail can vary

How are Tasks Learned?

- What does the user need to know?

- Do they need training?

  - academic

  - general knowledge / skills

  - special instruction / training

How are Tasks Learned (ATM machine)?

- Walk up & use system
  - can't assume much background/training
- Training?
  - too time consuming
- Must be simple & similar to existing systems
  - Drink machine

Where is the Task Performed?

- Office, laboratory, point of sale?
- Effects of environment on users?
- Users under stress?
- Confidentiality required?

- Do they have wet, dirty, or slippery hands?
- Soft drinks?
- Lighting?
- Noise?

Where (ATM machine)?

- Loud
  - dependence on voice I/O not a good idea
- Others looking over your shoulder
  - not private
  - PIN input must be confidential
    - do not confirm with sound
- Lighting is dim
  - make sure messages are readable

What is the Relationship Between Users & Data?

- Personal data
    - always accessed at same machine?
    - do users move between machines?
- Common data
    - used concurrently?
    - passed sequentially between users?
- Remote access required?
- Access to data restricted?

Data Relationships (ATM)

- Personal data
  - users may use any machine -> store on card
- Common data
  - fare rules (e.g., how much for transfer fee)
  - used concurrently
- Access to data restricted?
  - only you can use your ATM or credit card
- No need for remote access

What Other Tools Does the User Have?

- More than just compatibility
- How user works with collection of tools
  - example: automating lab data collection
    - how is data collected now?
    - by what instruments and manual procedures?
    - how is the information analyzed?
    - are the results transcribed for records or publication?
    - what media/forms are used and how are they handled?

Other Tools (ATM machine)

- Not relevant

How do Users Communicate With Each Other?

- Who communicates with whom? About what?

- Follow lines of organization? For/against it?

- Example: assistant to manager

  - installation of computers changes communication between them

  - people would rather change their computer usage than their relationship [Hersh82]

How do Users Communicate (ATM machine)?

- Not relevant

How Often do Users Perform the Tasks?

- Frequent users remember more details
- Infrequent users may need more help
    - even for simple operations
- Which function is performed
    - most frequently?
    - by which users?
    - optimize system for these tasks will improve perception of good performance

How Often (ATM machine)?

- Varying frequency of users
    - some take money out every day (most)
    - some take it only occasionally
- Varying frequency of tasks
    - can only do money transfer every 2 weeks
        - not frequent –> more instructions
    - might do put money in account every week
        - probably more common
- Observe users to find out for sure

What are the Time Constraints on the Task?

- What functions will users be in a hurry for?

- Which can wait?

- Is there a timing relationship between tasks?

Time Constraints (ATM machine)?

- Users will almost always be in a hurry
- Lines form
- Take less than 1 minute/transaction
- Be able to do any task in any order

What Happens When Things Go Wrong?

- How do people deal with
    - task-related errors?
    - practical difficulties?
    - catastrophes?
- Is there a backup strategy

Things Go Wrong (ATM machine)?

- Confusion on task
  - "dismiss transaction" button
- Practical difficulty
  - Change languages?
  - cash-in policy?
- Backup strategy

Selecting Tasks

- Real tasks users have faced
    - collect any necessary materials
- Should provide reasonable coverage
    - compare check list of functions to tasks
- Mixture of simple & complex tasks
- When you move to rough design stage
    - discard features that don't support your tasks
    - add a real task that exercises that feature

What Should Tasks Look Like?

- Say what the user wants to do, but not how the user would do it
  - allows comparing different design alternatives
- They should be very specific
  - forces us to fill out description with other details that become relevant
    - example: file browser story
- Some should describe a complete job
  - forces us to consider how features work together
    - example: phone-in bank functions

Look of Tasks (cont.)

- See where inputs come from and where outputs go
  - working with other tools
  - saving / loading
- Tasks should say who the users are
  - design can really differ depending on who
  - if possible, name names
    - allows getting more info. as it becomes relevant
  - characteristics of the users
    - job, expertise, etc.

Look of Tasks (cont.)

- Reflect interests of potential users

  - illustrate proposed functionality in context of work users really want to do

- Users are not always right

  - cannot anticipate new technology accurately

- Job is to build system users will want

  - not system users say they want

  - be very careful about this (you are outsider)

   - if you can't get users interested in your hot idea, you're probably missing something

Using Tasks in Design

- Write up a description of tasks
  - formally (book) or informally (us)
  - run by users and rest of the design team
  - get more information where needed
- Rough out an interface design
  - major screens & functions (not too detailed)
  - hand sketched
- Produce scenarios for each task
  - what user has to do & what they would see
  - step-by-step performance of task

Scenarios

- Scenarios are design specific
- Tasks are not design specific
- Scenarios force us to get specific
  - how various features will work together
  - settle design arguments by seeing examples
- Show users storyboards
  - sequences of sketches showing screens
  - actions users can take

Storyboards
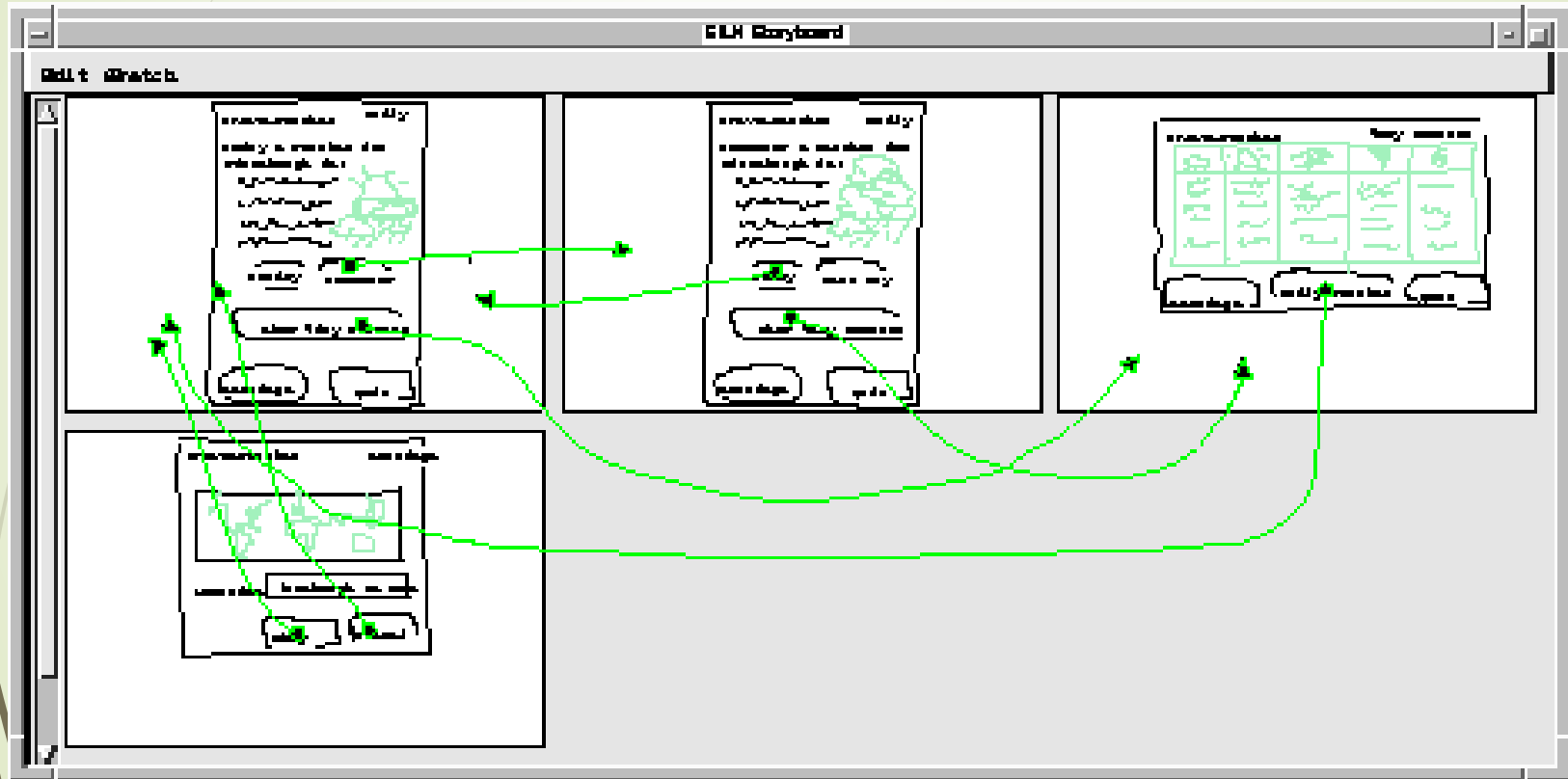
Cognitive Walkthrough

Evaluates a design for ease of learning

especially via exploration

Requires fairly detailed description of prototype

Analogy to code walkthrough

Taken from Abowd – Cognitive Walkthrough Lecture Note, Berkeley university

Watch VDO: ATM

Cognitive Walkthrough

- A task-oriented method of evaluating an interface *without* users

- A systematic way to imagine users' thoughts and actions when they use an interface for the first time.

- Benefits of evaluation before user meetings

  - Helps get rid of obvious problems that would waste users' time

  - May catch problems that testing with a few users will miss

Best Approach

- Tell a Believable Story
  - How does the user accomplish the task, action-by-action?
  - Based on user knowledge and system interface
- Work as a group
  - don't partition the task
- Be highly sceptical
  - remember the goal!
- Every gap is an interface problem

Cognitive Walkthrough How To - I

- Interface prototype
  - Define inputs
- Task description
  - Walk through action sequences for task
- Scenario – written list of the actions to complete the task in the interface
  - Record critical information
    - believability story
- An idea of who the users will be and their characteristic (so you can tell believable stories)
  - Personas may be useful

Inputs

Define interaction tasks

Identify users

what knowledge & experience

Prototype

Decompose tasks into action sequences

must know how interface looks for each step

Doing the walkthrough

Address each step of task sequence in turn

Formulate a believability story

answer 4 questions

Cognitive Walkthrough How To - II

- For each action in the sequence
  - tell the story of why the user will do it
  - ask critical questions (recall 7 Stages of Action)
    - *Will users be trying to produce the effect? I.e., will they form the goal designers wanted them to?*
    - *Will users see the correct control?*
    - *Will users recognize that this is the control they're after, i.e., that it will advance them toward their goal?*
      - *Or will they select a different control instead?*
    - *Will users understand the feedback?   That is, will be they be able to tell that they achieved their intended goal or at least made progress toward it?*

Question 1

- *Will the user be trying to produce whatever effect the action has?*

- Principle of Rationality applies here
  - There is a reason why people do things
  - Will user intent match intent for action?

Common supporting evidence

It is part of their original task.

They have experience using the system.

The system tells them to do it.

No supporting evidence?

Construct a failure story.

Question 2

Will the user be able to notice that the correct action is available?

Common supporting evidence

Known through experience

Visible device, such as a button

Visible representation of an action, such as a menu entry

Question 3

Once the user finds the correct action at the interface, will she know that it is the right one for the effect she is trying to produce?

Common supporting evidence

Experience

The interface provides a prompt or label that connects the action to what she is trying to do.

All other actions look wrong .

Question 4

After the action is taken, will the user understand the feedback given?

Common supporting evidence

Experience

Recognizing a connection between a system response and what she was trying to do.

Believability story

1. *Will the user be trying to produce whatever effect the action has?*

2. *Will the user be able to notice that the correct action is available?*

3. *Once the user finds the correct action at the interface, will she know that it is the right one for the effect she is trying to produce?*

4. *After the action is taken, will the user understand the feedback given?*

Watch VDO: iTunes Cognitive walkthrough

Benefits of a Cognitive Walkthrough

- Focus most on first experiences - learnability
- Easy to learn
- Can do early in the software cycle
- Surfaces and examines assumptions about what users might be thinking
- Can identify controls that are obvious to the designer but not to the user
- It can suggest difficulties with labels and prompts
- It can help find inadequate feedback
- Can help find inadequacies in the spec

Walkthrough Cons

- Is diagnostic, not prescriptive
- Focuses mostly on novice users
- Designers think like users
- Focus specifically on task-related issues
- Does not provide quantitative results

- A useful tool in conjunction with others

Walkthrough Example

- I have a library book that needs to be returned today. To help me remember, I want to set a reminder on my PDA. The reminder should display and beep at 5:00pm to remind me to return the book.

- Let's walkthrough this task on my PDA and identify usability issues, if any

Walkthrough Example

- Will a user try to produce the effect that the action has?

- Will a user see the control for the action?

- Will a user see that the control produces the desired effect?

- Will a user select a different control instead?

- Will a user understand the feedback to proceed correctly?

User Interface Models

A user interface is a medium of communication between a user and an application. The success of an application depends on the ease of its use. You may develop a very efficient application but if its interface is inefficient, the entire application is bound to fail. Four models that form the basis of the GUI design paradigm are:

User model: The end-user creates a user's model or a mental view that is the perception of the user's requirements.

Programmer model: The application developer understands the requirements explicitly stated by the user, analyses them, and creates a programmer model.

Design model: The programmer model is converted into a design model using various software engineering techniques.

Prototype model: Finally, the design model is implemented to result in a system image or the prototype.

Prototyping

This approach suggests building a prototype before developing the actual product. Prototyping is an iterative approach for development. Prototyping is part of the design phase of the product development. A prototype is a model or mockup of a system or device, used for evaluating a design. Prototypes are built using several shortcuts. This approach can be used for demonstrating feasibility of certain approach to solve a problem. After acceptance of the prototype, the development can be done by using a simple waterfall model of HCI design. Any part of a system could be prototyped, but the method is particularly valuable for user interface design.

What is a prototype in Interaction Design?

Low-fidelity

High-fidelity

- a series of screen sketches
- a storyboard, i.e. a cartoon-like series of scenes
- a lump of wood (e.g. PalmPilot)
- a cardboard mock-up
- a Powerpoint slide show
- a video simulating the use of a system
- a piece of software with limited functionality written in the target language or in another language

Why prototype?

- Evaluation and feedback are central to interaction design
- Stakeholders can see, hold, interact with a prototype more easily than with a document
- Team members can communicate effectively
- You can test out ideas for yourself
- It encourages reflection
- Prototypes answer questions, and support designers in choosing between alternatives

Watch VDO: paper prototype game design

What to prototype?

- Technical issues

- Work flow, task design

- Screen layouts and information display

- Difficult, controversial, critical areas

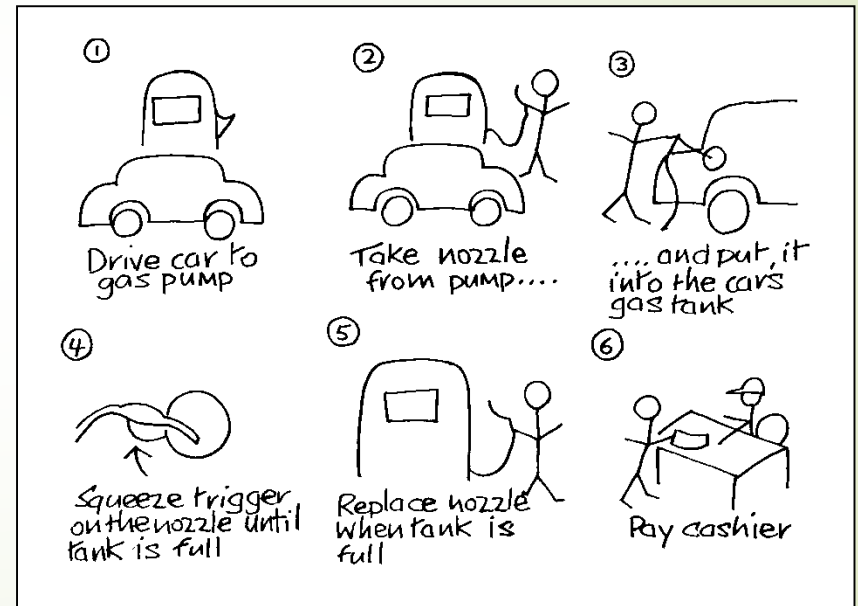# Types of prototypes

- Different kinds of prototyping
  - Low fidelity
  - High fidelity
    - Evolutionary
    - 'Throw-away'

Low-fidelity Prototyping

- Uses a medium which is unlike the final medium, e.g. paper, cardboard
  - Sketches of screens with post-it notes (task sequences)
    - Card-based
  - Storyboards
  - 'Wizard-of-Oz'
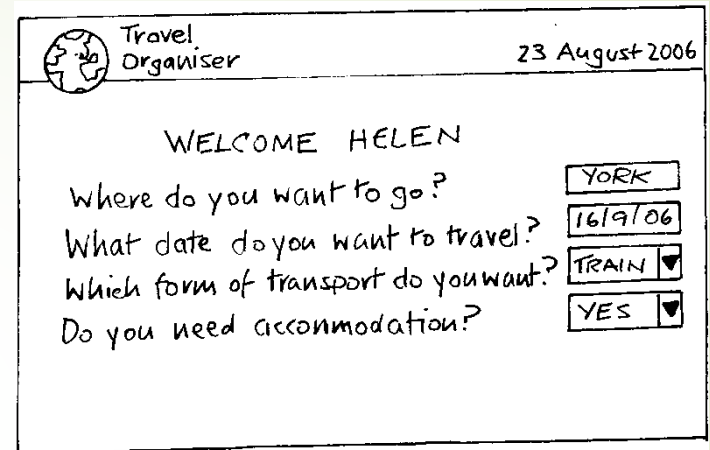- Quick, cheap, easily changed

- Watch VDO: Lo-fi vdo

Sketching

- Sketching is important to low-fidelity prototyping

- Don't be inhibited about drawing ability (even I sketch regularly)

Card-based prototypes

- Index cards (3 X 5 inches)
- Each card represents one screen or part of screen
- Often used in website development
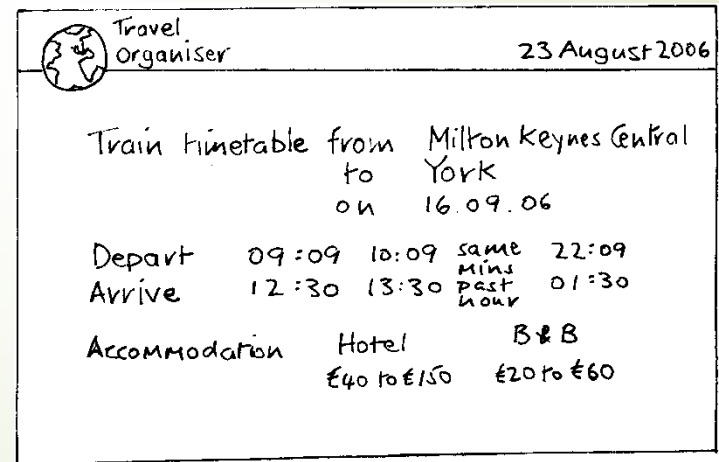

Travel Organiser   23 August 2006

WELCOME HELEN

Where do you want to go?   YORK
What date do you want to travel?   16/9/06
Which form of transport do you want?   TRAIN
Do you need accommodation?   YES

Travel Organiser   23 August 2006

Train timetable from   Milton Keynes Central
                to   York
                on   16.09.06

Depart   09:09   10:09   same mins   22:09
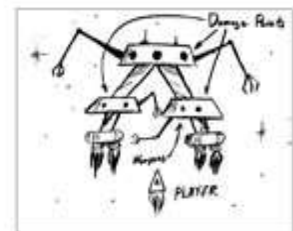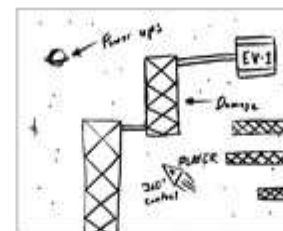Arrive   12:30   13:30   past hour   01:30
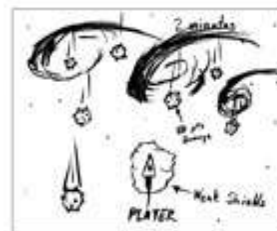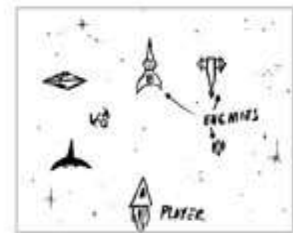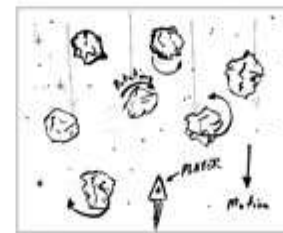
Accommodation   Hotel   B & B
                £40 to £150   £20 to £60

Watch VDO: paper prototype testing

Storyboards

➤ Often used with scenarios, bringing more detail, and a chance to role play

- It is a series of sketches showing how a user might progress through task using the device

- Used early in design
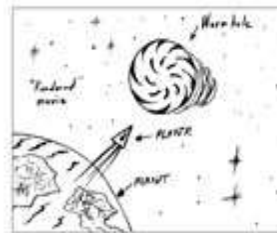
'Wizard-of-Oz' prototyping

➡ Users thinks they are interacting with a computer, but a developer is responding to output rather than the system.

◼ **What could the shortcomings of this approach be?**

User

>**Blurb blurb**
>**Do this**
>**Why?**

Watch VDO: Ex A Sketch wizard & Evaluating Speech
         Evaluating speech-enabled multimodal dialogue systems
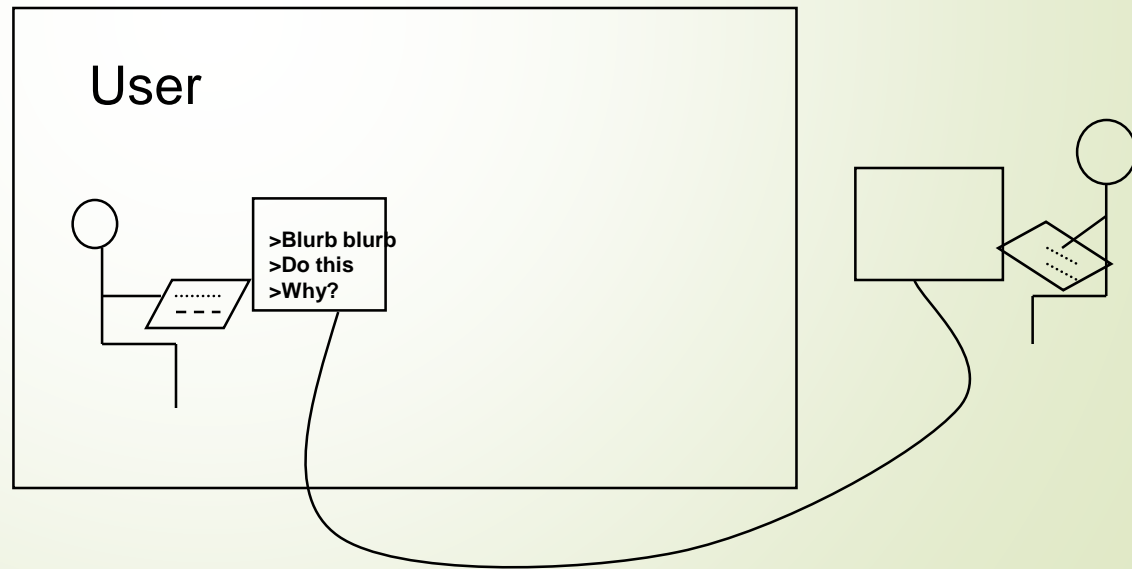
High-fidelity prototyping

- Uses materials that you would expect to be in the final product.

- Prototype looks more like the final system than a low-fidelity version.

- Common prototyping environments include Macromedia Director or Flash.

- Danger that users think they have a full system

Director File Edit View Insert Modify Control Xtras Window Help

Fri 11:26

Basic_finished: Stage (100%)

Stage (100%)

Trio Motor Company

Go to Sound and Video   Go to Animation

100 %

Basic_finished: Script

Script:Behavior Script 15:GoAnim

GoAnim   15   Internal

Lingo   mouseUp

```
1  on mouseUp me
2    _movie.go("Animation")
3  end
```

Property Inspector

Macintosh HD:Applications:Mac

Guides Movie Display Template

Stage Size 500   X 330

Channels: 150

Color: 10

Palette: System - Win

RGB   Index

Enable Edit Shortcuts

Preferred 3D Renderer:

Auto

Active 3D Renderer: #openGL

About:

Copyright:

Created By: j a -
Modified By: N/A - N/A
Font Map: Save   Load

Basic_finished: Cast

Cast:Internal

GoAnim

Trio Motor Compa   Go to Animation   Go to Sound and Vid   Backgroun
TitleText   Animation   SoundVide   Backgroun

tire   Trio   Go to Start   walkaroun
tire   Trio   GoStart   track1

Play   Pause   Rewind   track1

on exitFram e me   on exitFram e me   on mouseUp me   on mouseUp me
loop   AnimLoop   GoAnim   GoSndVid

on mouseUp me   on mouseUp me   on mouseUp me   on mouseUp me
GoToStart   RewindScri   PlayScript   PauseScrip

ping   22   23   24

ping

Code:Library

Library Behavior Inspe Object Inspect

Components

Components
  Button
  CheckBox
  DVDController
  DateChooser
  Label
  List
  NumericStepper
  RadioButton
  ScrollPane
  TextArea
  TextInput

Basic_finished: Score

Score

Start   Animation   Soun

Otrack

Member

1 5 10 15 20 25 30 35 40 45 50 55

1 TitleText   Background   walka
2 Trio   tire   tire   Play
3 Animation   Pause
4 SoundVideo   Rewin
5   SoundVideo
6   GoStart
7   Anim
8
9

Design:Text Inspector

High-fidelity Prototyping (Cont)

- **Evolutionary prototyping** - Evolutionary prototyping can be used when requirements either are understood poorly or are unknown. The prototype is written in the language to be used for the final product; after testing, the prototype is 'tidied up' and converted into the final product.
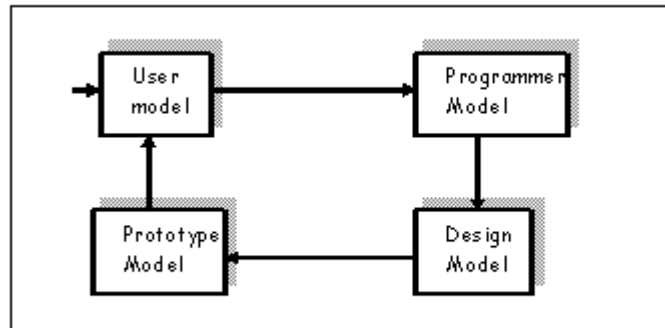

Figure 4.3 The combination of user interface models in the real world.

High-fidelity Prototyping (Cont)

▶ **Throw-away prototyping** – Throughway prototyping is used when requirements are modeled/understood poorly. The real system is developed quite separately using proper software engineering methods. Identification of unknown requirements is a secondary benefit. These prototypes can be adopted for reducing the cost and meeting schedule.

Compromises in prototyping

- Vertical: offers a small high fidelity and highly functional part of the system.

- Horizontal: represents the whole system in a cursory way – perhaps a low fidelity paper prototype (GUI)

Prototyping Summary

- Prototyping enables designers to build designs iteratively and involve users

- The most important thing to design is the users' conceptual model

- Different kinds of prototyping are used for different purposes and at different stages

  - Low fidelity at early stages for initial design solution attempts

  - High fidelity later to refine design solution

Research paper: **Using prototypes in early pervasive game development**
http://dl.acm.org/citation.cfm?doid=1371216.1371220

Card Sorting

Why do we sort things?

- If you can't find it, you can't "use" it
  - Libraries
  - Grocery stores
  - Databases
  - Etc.
- It makes maintaining and updating much, much easier.
  - Standardization (LOC/Dewey)

Research paper: **Card sorting activities with preschool children**
http://dl.acm.org/citation.cfm?id=1671011.1671036

How do we sort things?

- Standardization
  - Alphabetical
  - Numerical
  - Dewey/LOC
- Classification
  - Scientific (Aristotle, Linnaeus)
  - Taxonomy (the science of classification)

So, how do we find out what "fits"?

- Find out what peoples mental models are
  - An explanation in someone's thought process for how something works in the real world
  - Based on past experiences and prior knowledge
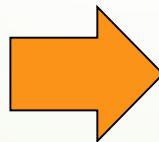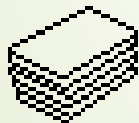 - Deeply ingrained and often subconscious

  Card sorting is an ideal way to discover peoples mental models

What is Card Sorting?

A technique for exploring users' mental models

How do you do it?

Start with a stack of cards
(one item per card)

Users sort cards into groups that make sense to them

How does card sorting work?

After you select (or recruit) a group of participants who closely resemble your user population, you should:
1. Give each participant (or two participants working together) a set of index cards. Each card should include one topic from your Web site.
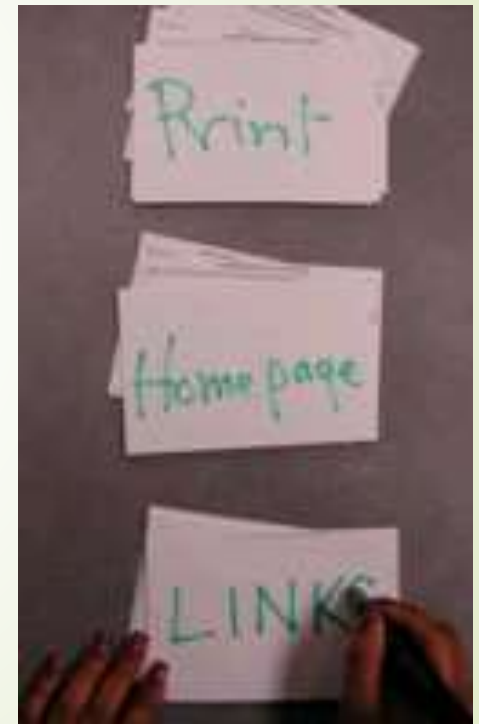
How does card sorting work?

2. Ask participants to group the cards in a way that makes sense to them. Many participants start by placing the first card on the table and then look at the second card to see whether it belongs in the same group or if it deserves its own category - and so on through the set of cards.

How does card sorting work?

3. After participants have grouped the cards, you can ask them to name or label each group.

How does card sorting work?

Watch VDO: card sorting

Why you should do it

- It's fast
- It's cheap
- It's flexible/portable
- It provides results
- It elicits user feedback

A bit more description…

- Card Sorting is typically described as:

    An activity where people are asked to group cards into either pre-named categories, or into unnamed groupings which are then named by the person

  - Open card sorting

  - Closed card sorting

  - Combination card sorting

Open Card Sorting

➥ The person creates and then names each pile of cards

When to do it:

➥ When you are trying to determine patterns in how your user groups organize their mental models (brand new site, etc)

An open card sort is typically done when you want to learn how users group content and understand the terms or labels users call each category.

Closed Card Sorting

- The person organizes cards into pre-named categories

When to do it:

- When you are trying to evaluate an existing IA, or confirm a proposed IA

- A closed card sort typically works best when you are working with a pre-defined set of categories and you want to learn how users sort content items into each category.

Combination Card Sorting

- A combination of open and closed card sorts

When to do it:

- Validate a previously conducted open card sort
- Prioritizing cards underneath the categories of a closed card sort

A closed sort works well after an open sort. By conducting an open card sort first, you can begin to identify categories of content. You can then use a closed card sort to see how well the category labels work.

After the sorting is done…

1. Record the results
   - Spreadsheet
   - Draw a picture of the categories (see proximity and organization of the piles)
2. Analyze the results
   - Look for patterns
   - Cluster analysis
3. Incorporate results into design

Focus Groups

www.uiweb.uidaho.edu/extension/focus.ppt

- Qualitative Research

- Concentrates on words and observations of individuals

- Obtain views, attitudes and ideas

Group discussion, conducted several times until similar trends and patterns in perception, attitudes, ideas are shown

The purpose of a focus group is to *promote disclosure*

A focus group measures:

- Community needs and issues;
- Citizens' attitudes, perceptions and opinions on specific topics;
- Impacts of a particular program on individuals and communities.

Advantages of a focus group:

- Easy to set up;
- Fast and relatively inexpensive;
- Can bring project personnel and beneficiaries together;
- Stimulates dialogue and new ideas;
- Generates ideas for evaluation questions to be included in other survey methods.

Advantages of a focus group:

• Socially oriented "synergism" (people respond in natural situations);

• Format allows the moderator to probe (flexible);

• High face validity (credible questions result in easily understood quotes and comments).

Disadvantages of a focus group:

- Easily misused
- Requires special moderator skills
- Data interpretation is tedious
- Avoiding bias can be difficult
- Capturing major issues can be difficult
- Results may not apply to the target population

Disadvantages of a focus group:

- Less control than a group interview
- Difficult to analyze (avoid lifting comments out of context)
- Focus groups vary (talkative, quiet, dull)
- Difficult to assemble, recruiting
- Discussion must be in an environment conducive to conversation

Types of focus group:

- Exploratory Focus Groups
  - Problem definition
  - Hypothesis generation
  - Concept testing
  - Pilot testing
- Experiencing Focus Groups
  - Product usage attitudes
  - Product usage beliefs

Thinking about a focus group?

- Consider your purpose for conducting focus group interviews

- Identify the users of the information generated by the focus group

- Develop a tentative plan including time required and resources needed

Recommended uses for a focus group:

- Planning needs assessment
- Planning program design
- Generate information for questionnaires
- Pilot programs
- Testing programs currently used
- Follow-up of a mail survey

Misuses for a focus group:

• Soliciting feedback about a product they have not actually tried

• Testing individual performance in a group setting

• Building consensus or resolving conflict

• Generating detailed information

Developing a plan:

- Audience to target
- Moderator and assistant
- Procedures to follow
- Time line
- Proposed budget

Participants in a focus group:

- Individuals with common characteristics (age,

interest, profession, etc.)

- Unfamiliar to each other (familiarity tends to inhibit

disclosure)

- Commonality, not diversity

Recruiting participants:

- Recruit with the purpose of the group in mind
- Identify the target group as carefully as possible
- Do not have spouses in the same focus groups
- Don't have superiors and subordinate employees in the same focus groups
- Seek participants through membership lists, directories, and associations

Recruiting participants:

- Check community activities before setting a time and date for the focus group

- Call potential participants 10-14 days before

- Send personalized invitations one week before

- Telephone each individual the day before the group

- Emphasize the benefits of attending and participating in a focus group

Designing questions:

- Questions are carefully determined, sequenced

- Moderator uses predetermined open-ended questions

- Questions appear spontaneous but are carefully designed and in a logical sequence and *memorized* by the moderator

Designing questions…

- Use open-ended questions

- Avoid dichotomous questions

- Don't ask "why"

- Use "think back" questions

Designing questions…

- Prepare questions carefully

- Ask "uncued" questions first

- Consider standardized questions

- Focus the questions: general to specific

- Be cautious of "what if" questions

"Ending" questions…

- All things considered… reflective

- Summary question… did we hear you?

- Final question… verify completeness

Moderating the focus group:

• Identify a trained moderator and an assistant to conduct the focus group interview.

• The moderator creates a warm and friendly atmosphere, directs and   keeps the flow of the conversation and takes notes.

Moderating the focus group:

Key success factor:

Ability of moderator to stimulate INTERACTION

among group members

Before the session you will need:

- Comfortable room for conducting focus group
- High quality tape or video recorder
- High quality microphone
- Check equipment before the session
- Cassette audio or video tapes

Before the session you will need:

- Notepad for taking notes
- Name tags for first names only
- Form to receive mileage
- Educational material for handouts
- Refreshments if planned

Managing the Focus Group Session:

- Greet the participants as they arrive
- Hand out form to receive mileage or incentive
- Distribute educational material or handouts
- Hand out and collect demographic surveys

Managing the Focus Group Session:

- Deliver a prepared introduction
- Welcome
- Overview of topic
- Ground rules
- Answer questions
- First question

Immediately after the Focus Group:

- Draw a diagram of the seating arrangement
- Check performance of tape recorder
- Debrief moderator and assistant

Analyzing the Focus Group Session:

- Begin while group is in session
- Within hours, analyze individual group
- Within days, analyze series of groups
- Prepare draft report, circulate
- Finalize report

Analysis considerations:

- Consider the words
- Consider the context
- Consider internal consistency
- Consider frequency of comments
- Consider intensity of comments
- Consider specificity of comments
- Find the big ideas

Watch VDO: Focus Group Research Salad Dressing

Research papers

- User Expectations for Mobile Mixed Reality Services: an Initial User Study
http://www.cs.tut.fi/~olsson/olsson_etal_UserExpectationsForMobileMixedRealityServices_anInitialUserStudy
(using focus group)

- Developing mobile mixed reality application based on user needs and expectations
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6207780&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D6207780

Usability Methods

- Personas
- Scenarios
- Task analysis
- Cognitive walkthrough
- User Interface Models
- Prototyping
- Card Sorting
- Focus group

References

- Norman, Donald, "The Design of Everyday Things", (2002)

- Shneiderman, Ben, "Designing the User Interface, Strategies for Effective Human-Computer Interaction", (2004)

- Lecture 5: Prototyping, *www.informatics.sussex.ac.uk/courses/ihci/08/prototyping.ppt*

- Gibbs, A. 1997. Focus Groups. Social Research Update.   University of Surrey. (http://www.soc.surrey.ac.uk/sru/)

- Douglah, M. 2002. Focus Groups Workshop. U Wisc.   Extension. (http://www.uwex.edu/ces/pdande)

- Morgan D.L. 1997. Focus groups as qualitative research, 2nd Ed. London: Sage.

- Templeton, Jane Farley. 1996. The Focus Group: A Strategic Guide to Organizing, Conducting and Analyzing the Focus Group Interview (revised edition) McGraw Hill Trade.

- Krueger, Richard A. 1997. Developing Questions for Focus Groups. London. Sage .

Individual Work II (Revisit)

- Design a new application or device

- Examine how it has been designed, paying attention to how the user is meant to interact with it?

  - Write down:

    1. Its main user groups

    2. Its functionality

    3. Explain what are the most important usability and user experience measurement for your web/application

       - Select 2 from each goals and principles and discuss the reason in detail

    4. Use them to assess how well? Describe the possibility of how would you measure those goals from users during evaluation process.

       - Eg. user experience goals :Is it fun to use? > Measure heart beat per min

         Usability goals: What specific mechanisms have been used to ensure safety? > Count files lost in 1 day

  - Repeat 3 for Design principles and Usability principles