

Användbarhetsutvärdering och inspektion



Varför?

- Mäta användbarhet
- Objektiv/subjektiv återkoppling
- “Quick and dirty” vs. “Slow and clean”
- Med eller utan användare
- Iterativ design
- Prototyping

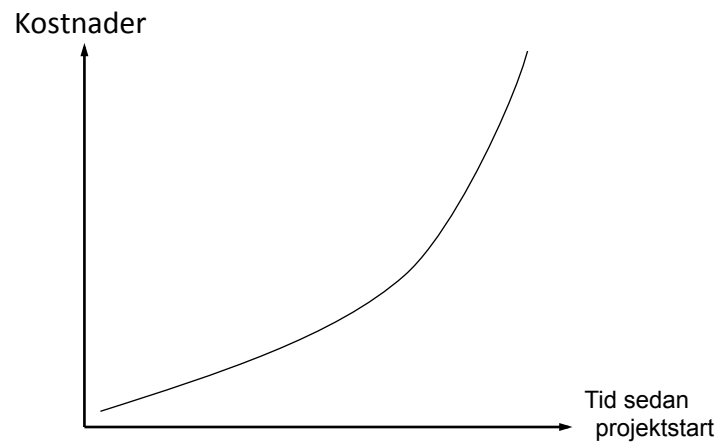




Användbarhetsutvärdering

- Användbarhet kan mätas genom sådana frågor
- Användbarhetsutvärdering bygger på analys och experiment
- No "silver bullet"
- Utvärdering handlar typiskt om huruvida systemet uppfyller kraven, det förklarar sällan orsakerna bakom problemen.
- Utvärdera under utvecklingen, inte efteråt – användbarhet är inte bara ett mått utan ett mål för utvecklare och för utveckling
- Risk att den oerfarne utvecklaren ignorerar validering till förmån för verifiering av system

Kostnader för att åtgärda fel



Formativ eller Summativ utvärdering

- **Formativ utvärdering**
 - Hjälper till att forma systemet
 - Iterativ testning
- **Summativ utvärdering**
 - Testar hela systemet på slutet (när det är för sent att göra något)

Formativa utvärderingar

- Erfarenheten hjälper dig att bygga prototyper och att fatta konkreta beslut
- Informella användartester med enkla prototyper mäter fel tidigt och enkelt.
- Fältextperiment med prototyper med en serie av förändringar för att mäta skillnader i preferens, felfrekvens, hastighet, etc.
- Kontrollerade experiment (alla viktiga parametrar är under kontroll)

Empirisk eller Analytisk utvärdering

- Empirisk utvärdering
 - Låt användare prova en prototyp
 - Enkla, billiga tester
 - Kvalitativa resultat, men ofta av avgörande betydelse
- Analytisk utvärdering
 - Använder teoretiska modeller för att resonera runt användningen
 - Rigorösa och kostsamma tester
 - Kvantitativa, jämförbara resultat

Usability testing & research

Usability testing

- Improve products
- Few participants
- Results inform design
- Usually not completely replicable
- Conditions controlled as much as possible
- Procedure planned
- Results reported to developers

Experiments for research

- Discover knowledge
- Many participants
- Results validated statistically
- Must be replicable
- Strongly controlled conditions
- Experimental design
- Scientific reported to scientific community

Performance measurement

- Görs normalt i labbmiljö
- Kvantitativa data/Objektiva resultat
 - Lätt att jämföra
- Framför allt intresserad av
 - om användaren kan genomföra en uppgift
 - i så fall hur snabbt det går
 - vilka fel gjordes

Performance measurement

- Identifiera mätbara användbarhetsmål. Ta fram användarprofiler.
- Ta fram manus och formulär för utvärderarna.
- Genomför studien. Låt testanvändarna gå igenom de förutbestämda uppgifterna. Samla in data, "task completion, errors and times"
- Analysera data och statistik
- Rapport med rekommendationer. Videoklipp.

Testing conditions

- Usability lab or other controlled space.
- Emphasis on:
 - selecting representative users;
 - developing representative tasks.
- 5-10 users typically selected.
- Tasks usually last no more than 30 minutes.
- The test conditions should be the same for every participant.
- Informed consent form explains procedures and deals with ethical issues.

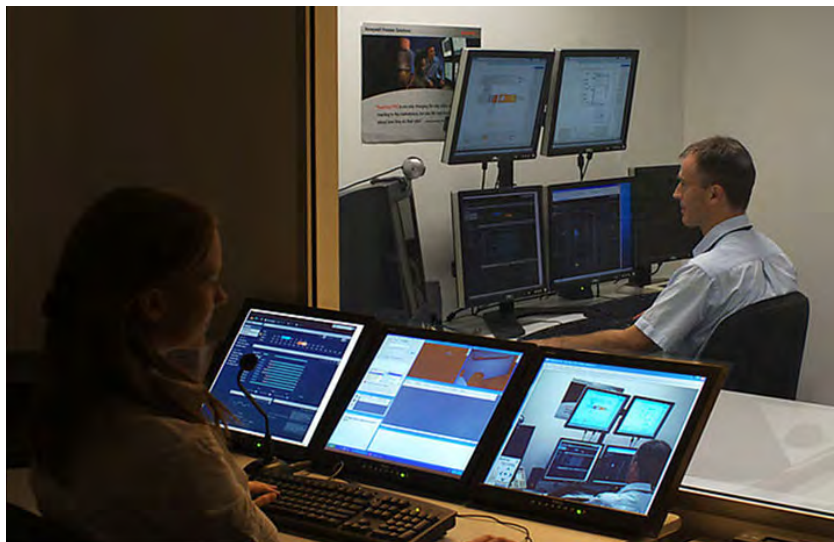
Some type of data

- Time to complete a task.
- Time to complete a task after a specified. time away from the product.
- Number and type of errors per task.
- Number of errors per unit of time.
- Number of navigations to online help or manuals.
- Number of users making a particular error.
- Number of users completing task successfully.

Användbarhetslaboratorium där observatörer betraktar en användare och en hjälpreda



Loggning, inspelning, direktanalys



Bärbar utrustning för användbarhetsarbete i fält med mobil utrustning



Performance measurement



Videokamera, mikrofon, scan converter, envägsspegel, mjukvara för loggning

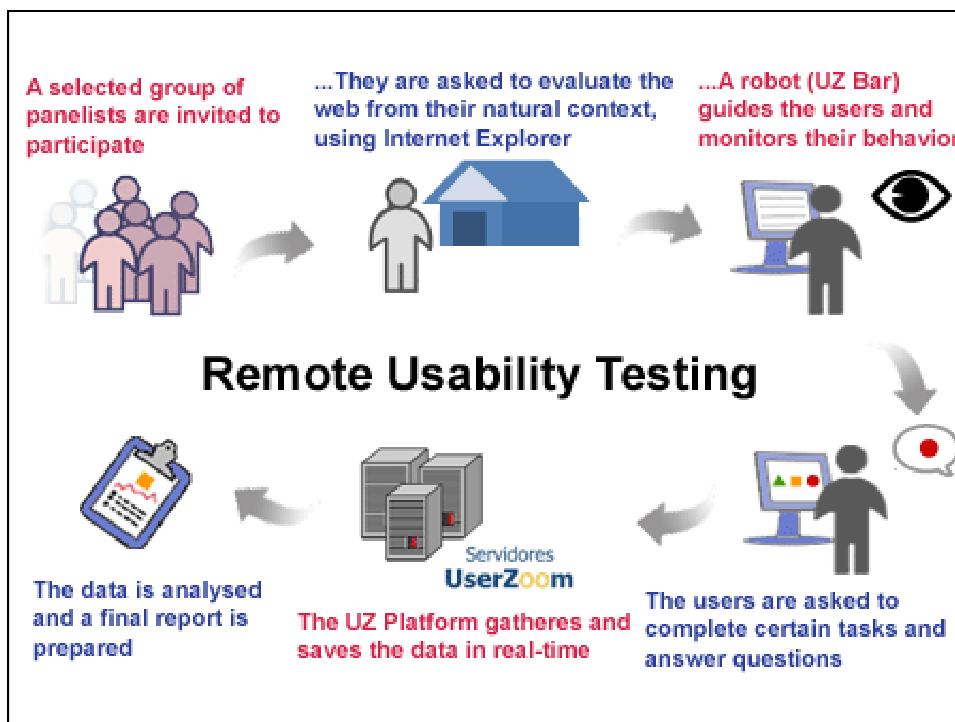
Effektivt för att övertyga utvecklare om bristande användbarhet



Developer watching videotape of usability test.

Fördelar – Nackdelar

- | | |
|---|--|
| <ul style="list-style-type: none"> + Kvantitativa data är lätta att jämföra + Kvantitativa data är bra när man vill få tyngd bakom ett argument t.ex. "3 av 12 personer lyckades inte registrera sig på sajten" + Bra när man vill jämföra olika designlösningar | <ul style="list-style-type: none"> – Smakar det så kostar det. <ul style="list-style-type: none"> - användare, lab, personal, tid – Inga kvalitativa data. Missar subjektiva data. Kan vara mycket bra vid omdesign. – Kräver erfaren personal – Erfarna användare krävs för yrkesmässigt beteende |
|---|--|



Användbarhetsutvärdering med eye-tracking



Användbarhetsutvärdering med eye-tracking



Användbarhetsutvärdering med eye-tracking



Thinking Aloud

- Görs med användare. I labbmiljö eller hos användarna
- En användare i taget gör förutbestämda uppgifter
- Användarna verbaliserar sina tankar – ombeds tänka högt när de använder systemet
- Kvalitativa data
- Identifierar missförstånd i systemet

Exempel på thinking aloud från Inuse



Metoden Thinking Aloud

- Identifiera användbarhetsmål och användarprofiler.
- Planera testet. Ta fram protokoll etc.
- Genomför testet. Uppmuntra användarna att tänka högt. Samla in data. Vad gör dom – vad säger dom.
- Analysera data
- Rapport med rekommendationer. Videoklipp.

Thinking Aloud: Fördelar och nackdelar

- | | |
|---|--|
| + Fångar upp subjektiva data om gränssnittet | - Inga kvantitativa data, bara kvalitativa |
| + Fångar ofta upp intressanta aspekter som man inte hade tänkt på innan | - Vissa användare har svårt för att tänka högt – det är inte naturligt |
| + Identifierar många problem, effektiv metod | - När man utför svåra uppgifter kan det vara svårt att tänka högt |
| + Man får återkopplingen direkt om vad som är problematiskt | - Svårt att tänka högt om sitt beslutsfattande |
| + Ser vad de gör och kan jämföra det med vad de säger | - Mycket brus, gäller att kunna plocka ut det som är relevant av allt som sägs |

Enkäter

- Används för att kartlägga subjektiv tillfredsställelse
 - + Kan distribueras till många människor
 - Användare svarar vad de tror att de gör, inte vad de verkligen gör

Kortsortering

- Fysiska papperslappar med text och bilder
- Testpersoner sorterar bilderna i viktighetsordning
- Bra för att sortera ikoner, begrepp, etc.
- Kan förbättra användarnas relevans och förståelse för begrepp
- Kräver verkliga användare för bra resultat

Analytiska utvärderingsmetoder



Analytisk evaluering

- En logisk simulering av användarnas beteende
- T. ex. GOMS, KLM, Cognitive Walkthrough
- Kan användas för att testa saker innan de har byggts
- Sparar tid eftersom vi inte behöver bygga något i förväg
- Inga användarexperiment behövs
- Skapa en sekvens av steg och mät tiden per steg.

Predictive models

- Provide a way of evaluating products or designs without directly involving users.
- Less expensive than user testing.
- Usefulness limited to systems with predictable tasks - e.g., telephone answering systems, mobiles, cell phones, etc.
- Based on expert error-free behavior.

GOMS

Mäter hastigheten för att genomföra vissa handlingar

Beskriver olika möjliga vägar (sekvenser) för att lösa problemet

- Goal – det mål som ska uppnås
- Operator – möjliga handlingar
- Methods – Sekvens av steg
- Selection rules – regler för att välja bland de olika metoderna

GOMS modell

```

GOAL:  PHOTOCOPY PAPER
        GOAL:  LOCATE-ARTICLE
        GOAL:  PHOTOCOPY-PAGE repeat until no more pages
                GOAL: ORIENT-PAGE
                        OPEN-COVER
                        SELECT-PAGE
                        POSITION-PAGE
                        CLOSE-COVER
                GOAL: VERIFY-COPY
                        LOCATE-OUT-TRAY
                        EXAMINE-COPY
        GOAL: COLLECT-COPY
                LOCATE-OUT-TRAY
                REMOVE-COPY (OBS! Yttre målet fullföljt!)
        GOAL: RETRIEVE-JOURNAL
                OPEN-COVER
                REMOVE-JOURNAL
CLOSE-COVER
  
```

Keystroke level model

- GOMS has also been developed to provide a quantitative model - the keystroke level model.
- The keystroke model allows predictions to be made about how long it takes an expert user to perform a task.

Keystroke-Level Analysis

- Dela in varje uppgifts beteende i olika delkomponenter
- Tilldela varje komponent en exekveringstid
- T ex. Tangenttryckning – 0.12 s, peka med musen – 1.10 s
+ bra för att jämföra olika alternativa lösningsförslag
- fungerar bara för mindre delar av systemet

KLM model

Operation	Anmärkning	Tid (i sekunder)
K	Tryck ned tangent	0,12
	Good maskinskrivare (90 ord/min.)	0,28
	Medelmättig maskinskrivare (40 ord/min.) Icke-maskinskrivare	1,20
B	Musknappstryckning	0,10
	ner eller upp dubbelklick	0,20
P	Peka med musen Fitts lag Medelrörelse	$0,11\log(D/S+0,5)$ 1,10
H	Hand till eller från tangentbord	0,40
D	Rita (domänberoende)	-
M	Mentalt förbereda sig	1,35
R	Respons från systemet	-

Tabell X: tider för vissa operationer som en operatör utför enligt KLM-metoden (efter Card, Moran & Newell, 1983)

Response times for keystroke level operators (Card et al., 1983)

Operator	Description	Time (sec)
K	Pressing a single key or button	0.22
	Average skilled typist (55 wpm)	0.28
	Average non-skilled typist (40 wpm)	0.08
	Pressing shift or control key Typist unfamiliar with the keyboard	1.20
P	Pointing with a mouse or other device on a display to select an object. This value is derived from Fitts' Law which is discussed below.	0.40
P1	Clicking the mouse or similar device	0.20
H	Bring 'home' hands on the keyboard or other device	0.40
M	Mentally prepare/respond	1.35
R(t)	The response time is counted only if it causes the user to wait.	t

KLM exempel

- **Exempel.** Det finns två olika sätt att utföra kommandot "Spara". Antingen kan man välja Ctrl-S som en tangentkombination på tangentbordet, eller också kan man välja alternativet spara i arkivmenyn.
- Alternativ 1: Ctrl S – $T_1 = M + H + K = 1,35 + 0,40 + 0,28 = 2,03$ sek
- Alternativ 2: Menyalternativ – $T_2 = M + H + P + B + P + B = 1,35 + 0,40 + 1,10 + 0,10 + 1,10 + 0,10 = 4,15$ sek

KLM

- For most systems, there are 5 components:
 - K (0.2 s) – press a **key** or mouse button
 - P (1.1 s) – **p**oint with mouse
 - H (0.4 s) – **h**ome on keyboard, mouse or other device
 - M (1.35 s) – **m**entally prepare
 - R (t) – system **r**esponse time (needs to be measured)
- MHPKPK
 - That is a **m**ental preparation, followed by a '**h**ome' (movement from keyboard to mouse), followed by two pairs of **p**oints and clicks on the mouse.
 - 4.35s.

Fitts' Law (Fitts, 1954)

- Fitts' Law predicts that the time to point at an object using a device is a function of the distance from the target object & the object's size.
- The further away & the smaller the object, the longer the time to locate it and point to it.
- Fitts' Law is useful for evaluating systems for which the time to locate an object is important, e.g., a cell phone, a handheld devices.

Pluralistic walkthrough

- Variation on the cognitive walkthrough theme.
- Performed by a carefully managed team.
- The panel of experts begins by working separately.
- Then there is managed discussion that leads to agreed decisions.
- The approach lends itself well to participatory design.

Cognitive walkthroughs

- Focus on ease of learning.
- Designer presents an aspect of the design & usage scenarios.
- Expert is told the assumptions about user population, context of use, task details.
- One or more experts walk through the design prototype with the scenario.
- Experts are guided by 3 questions.

The 3 questions

- Will the correct action be sufficiently evident to the user?
- Will the user notice that the correct action is available?
- Will the user associate and interpret the response from the action correctly?

As the experts work through the scenario they note problems.

Cognitive Walkthrough

- Inga användare är involverade
- Istället simulerar *experter* användarnas aktiviteter. – En utvärderare går igenom systemet och försöker att spela en användare
- Handlingarna baseras på vår kunskap om användarna
 - + Quick
 - Dirty
 - Blandar inte in användare, svårt att fånga domänspecifika problem

Cognitive Walkthrough

Explanatory learning

- Konstruerad för nybörjaranvändare
- Lämplig för “walk-up-and-use-interfaces”
- En successiv genomgång av uppgifter baserat på frågeställningar på olika nivåer
 - Ser användaren de olika sökalternativen?
 - Väljer användaren rätt alternativ?
 - Förstår användaren systemets svar?

Fördelar Cognitive Walkthrough

- Billig metod. Inga användare.
- Kräver inte fullständig prototyp för att använda
- Fångar upp många fel, framför allt de som relaterar till inlärning

Nackdelar Cognitive Walkthrough

- Svårt att avgöra hur användare reagerar.
- Svårt när man har olika användargrupper
- Tar tid att genomföra
- Fungerar för vissa system men inte för andra.

Metod Cognitive Walkthrough

- Identifiera och beskriv typiska användare
- Identifiera och beskriv typiska arbetsuppgifter
- Gå igenom sekvensen av steg som användarna tar i gränssnittet för varje uppgift.
 - Will the correct action be sufficiently evident to the user?
 - Will the user notice that the correct action is available?
 - Will the user associate and interpret the response from the action correctly?
- Samla data. Antaganden. Noteringar.
- Rapportera. Redesign.

Heuristic Evaluation

- Jacob Nielsen
 - www.useit.com
- Metod för expertbedömning
- Checklista med heuristik
- Inga användare involverade
- Mycket lättanvänd
- Quick and Dirty
- Flera experter utvärderar



Heuristic evaluation

- Developed by Jakob Nielsen in the early 1990s.
- Based on heuristics distilled from an empirical analysis of 249 usability problems.
- These heuristics have been revised for current technology.
- Heuristics being developed for mobile devices, wearables, virtual worlds, etc.
- Design guidelines form a basis for developing heuristics.

Discount evaluation

- Heuristic evaluation is referred to as discount evaluation when 5 evaluators are used.
- Empirical evidence suggests that on average 5 evaluators identify 75-80% of usability problems.

Fördelar med Heuristic Evaluation

- Går snabbt att genomföra. Billig metod.
- Kan göras i olika faser i projektet. Kräver inte en interaktiv prototyp.
- Kan hitta potentiella problem tidigt.
- Gör användartest i senare skede mer kostnadseffektivt.

Nackdelar med Heuristic Evaluation

- Inga användare involverade.
- Kräver flera utvärderare.
- Beror väldigt mycket på de enskilda utvärderarnas kunskaper.
- Ger inte samma tyngd vid argumentation.
- Svårt vid domänrelaterade problem

Metod Heuristic Evaluation

- Bestäm vilka delar av applikationen som ska utvärderas.
- Bestäm om checklistan ska kompletteras eller modifieras.
- Gå igenom applikationen. Dokumentera potentiella problem för varje punkt i checklistan.
- Analysera och rapportera.

Nielsens Heuristics

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Help users recognize, diagnose, and recover from errors
6. Error prevention
7. Recognition rather than recall
8. Flexibility and efficiency of use
9. Aesthetic and minimalist design
10. Help and documentation

Nielsen's new design heuristics

- **Visibility of system status**

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

- **Match between system and the real world**

The system should speak the users' language, with words, phrases and concepts familiar to the user rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logic order

Nielsen's new design heuristics

- **User control and freedom**

Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

- **Consistency and standards**

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

Nielsen's new design heuristics

- **Error prevention**

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

- **Recognition rather than recall**

Make objects, actions and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Nielsen's new design heuristics

- **Flexibility and efficiency of use**

Accelerators – unseen by the novice user – may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

- **Aesthetic and minimalist design**

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Nielsen's new design heuristics

- Help users recognise, diagnose and recover from errors

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

- Help and documentation

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Heuristic evaluation

1. Simple and natural dialogue
2. Speak the users language
3. Minimise Users Memory Load
4. Consistency
5. Feedback
6. Clearly marked exits
7. Shortcuts
8. Good error messages
9. Prevent errors
10. Help and documentation

Simple and Natural dialogue

- Obvious how to interact with the system
- Grouping of objects
- Enough information on the screen
- Overview and details
- Avoid confusing decoration

Speak the users language

- Domain language
- Avoid system or computer language
- Use metaphors familiar to the users
- Icons, pictures, etc., relevant for the domain

Minimise user memory load

- Enough information on the screen
- Recognition instead of recall
- Highlight important information
- Show status information
- Default values

Consistency

- Consistent layout
- Consistent interaction
- Consistent language
- Consistent functionality

Feedback

- Show waiting times
- Feedback for actions
- Work related feedback
- System failure

Clearly marked exits

- Show the way the user has “walked” through the system
- How to get back and forward
- Oriented in the information space

Shortcuts

- Use shortcuts to perform actions
- Type ahead
- Jumps to a desired location
- Reuse of interaction history

Good error messages

- Avoid obscure data code
- “An error No 13 has occurred”
- Use the users language

Prevent error

- Dialogues for actions that may lead to serious consequences
- Avoid modes
- Show status
- Support “undo” and “redo”

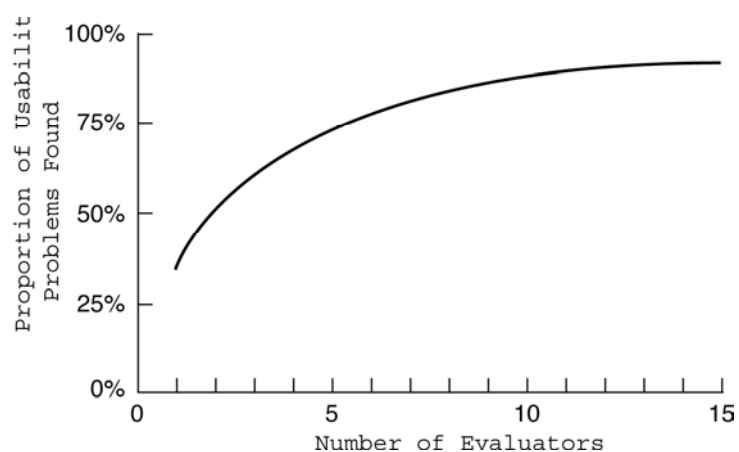
Help and documentation

- Manuals should not be necessary
- Give on-line help on the users initiative
- If no other solution is possible, give help at the initiative of the system

Heuristic evaluation

- Not suitable for repeated analysis (with the same person). Can be solved by rotating the inspection tasks
- Good results if the evaluators are skilled
- Good in combination with real user testing (the techniques complement one another)

No. of evaluators & problems



3 stages for doing heuristic evaluation

- Briefing session to tell experts what to do.
- Evaluation period of 1-2 hours in which:
 - Each expert works separately;
 - Take one pass to get a feel for the product;
 - Take a second pass to focus on specific features.
- Debriefing session in which experts work together to prioritize problems.

Advantages and problems

- Few ethical & practical issues to consider because users not involved.
- Can be difficult & expensive to find experts.
- Best experts have knowledge of application domain & users.
- Biggest problems:
 - Important problems may get missed;
 - Many trivial problems are often identified;
 - Experts have biases.

Grading the errors

- 0 I do not agree that this is a usability problem at all
- 1 Cosmetic problem only
- 2 Minor usability problem: should be given low priority
- 3 Major usability problem: important to fix, so should be given high priority
- 4 Usability catastrophe: imperative to fix this before product can be released

How to use the results

- Analyse the results
- Identify solutions to the problems
- Create a new prototype
- Optimising between requirements

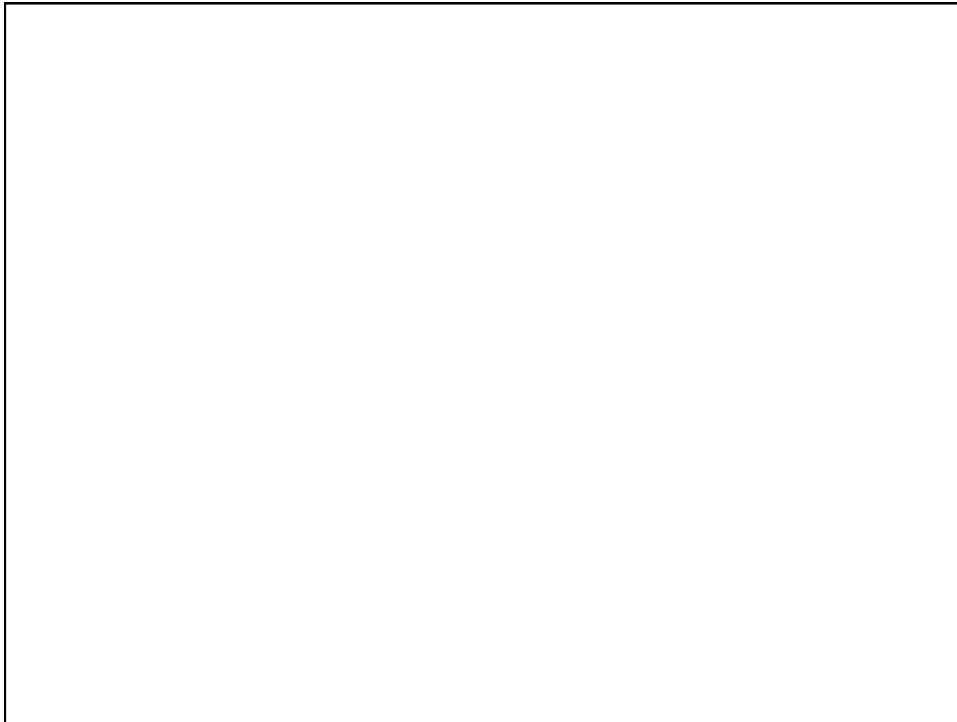
Evaluation steps

- Identify usability aspects, specify special requirements on the product
- Produce a prototype
- Prepare experiments (Find users, select parameters to test, documentation method)
- Test and collect data
- Analysis
- Conclusions

Evaluate...

Which method should you use and why?

- Micro wave oven - which is the fastest way of heating your food, 1.10 or 1.11?
- System for phoning via your computer.
- Public service phone (e.g. Bank, RSV)
- Web site for finding courses (e.g. Asken)
- Digital wrist watch with one button



Key points

- Expert evaluation: heuristic & walkthroughs.
- Relatively inexpensive because no users.
- Heuristic evaluation relatively easy to learn.
- May miss key problems & identify false ones.
- Predictive models are used to evaluate systems with predictable tasks such as telephones.
- GOMS, Keystroke Level Model, & Fitts' Law predict expert, error-free performance.

DECIDE: a framework to guide evaluation

- Determine the *goals*.
- Explore the *questions*.
- Choose the evaluation *approach and methods*.
- Identify the *practical issues*.
- Decide how to deal with the *ethical issues*.
- Evaluate, analyze, interpret and present the *data*.

Determine the goals

- What are the high-level goals of the evaluation?
- Who wants it and why?
- The goals influence the approach used for the study.
- Some examples of goals:
 - Identify the best metaphor on which to base the design.
 - Check to ensure that the final interface is consistent.
 - Investigate how technology affects working practices.
 - Improve the usability of an existing product .

Explore the questions

- *All* evaluations need goals & questions to guide them.
- E.g., the goal of finding out why many customers prefer to purchase paper airline tickets rather than e-tickets can be broken down into sub-questions:
 - What are customers' attitudes to these new tickets?
 - Are they concerned about security?
 - Is the interface for obtaining them poor?
- What questions might you ask about the design of a cell phone?

Choose the evaluation approach & methods

- The evaluation *approach* influences the *methods* used, and in turn, how data is collected, analyzed and presented.
- E.g. field studies typically:
 - Involve observation and interviews.
 - Do not involve controlled tests in a laboratory.
 - Produce qualitative data.

Identify practical issues

For example, how to:

- Select users
- Stay on budget
- Stay on schedule
- Find evaluators
- Select equipment

Decide about ethical issues

- Develop an informed consent form
- Participants have a right to:
 - Know the goals of the study;
 - Know what will happen to the findings;
 - Privacy of personal information;
 - Leave when they wish;
 - Be treated politely.

Evaluate, interpret & present data

- The approach and methods used influence how data is evaluated, interpreted and presented.
- The following need to be considered:
 - Reliability: can the study be replicated?
 - Validity: is it measuring what you expected?
 - Biases: is the process creating biases?
 - Scope: can the findings be generalized?
 - Ecological validity: is the environment influencing the findings? - i.e. Hawthorn effect.