Interação Pessoa-Máquina

Teresa Romão
tir_at_fct.unl.pt
DI/FCT/UNL

2022/2023

Main Objectives

- Understand the human factors which drive the usage of computer systems.
- Understand novel paradigms for human-computer interaction
- Know and apply usability principles.
- Know and apply prototyping techniques.
- Know and apply interfaces evaluation techniques.
- Develop creative capabilities to come up with innovative solutions for interaction problems.
- Fit HCI in the engineering project.

Program

- Introduction - Human-Computer Interaction (HCI): What? Why? When?
- Usability principles
- Characteristics of interactive systems
- Human factors in the HCI
- User centered design and iterative design process
- User and task analysis
- Sketching and prototyping
- Interaction design principles
- Visual design
- Evaluation methods
- Interaction styles and paradigms
- Future perspectives

Textbooks

- Gonçalves, D., Fonseca, M.J., and Campos, P., Introdução ao Design de Interfaces. FCA, 2017.
- Norman, Donald. *The Design of Everyday Things*. MIT Press, 1998.
- Nielsen, Jacob, *Usability Engineering*, Academic Press, 1993.
- Dix, Alan, Finlay, Janet, Abowd, Gregory, Beale, Russel. Human-Computer Interaction. Prentice Hall Europe, London, 2003.

Lecture slides will be available on CLIP.

Complementary:

- Buxton, B., Sketching User Experiences, Morgan Kaufman, 2007.
- Mullet, K. and Sano, D., Designing Visual Interfaces, Prentice Hall, 1995.
- Tufte, E. *Envisioning Information*, Connecticut Graphic Press, 2003.

Additional readings will be provided during classes and on the course web site.

Evaluation

Final grade =

35%T1 + 35%T2 + 30%PW

Minimal grades:

(mean (T1; T2) >= 9,5) AND (PW >= 10)

Evaluation

Dates (to be confirmed):

- T1: November, 2022
- T2: December, 2022

Mandatory lab classes:

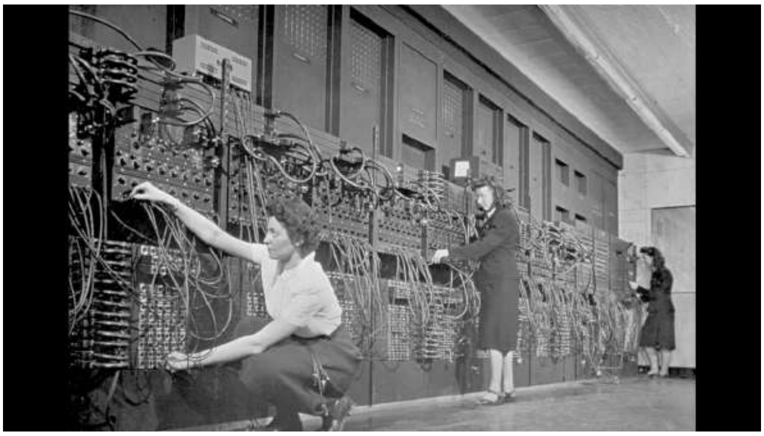
- Prototype testing day (October 18, 19 e 20) – to be confirmed
- Heuristic evaluation
- Project presentation

Course web site

http://ctp.di.fct.unl.pt/~tir/IPM

Interface Design

Interface Design



Interface Design

Interface Design

Interface Design

Interface Design

People shouldn't have to read a manual to open a door, even if it is only one word long (push/pull)

Don Norman

Interface Design

Interface Design

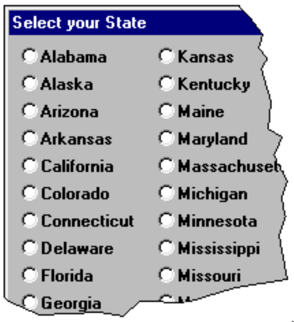
Interface Design

Interface Design

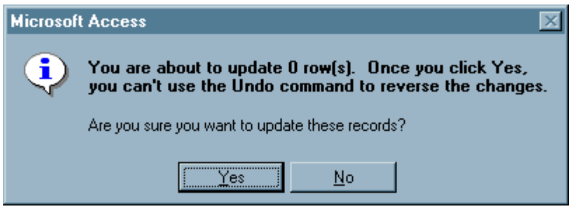
Alternative design by Benjamin Bannister

Interface Design

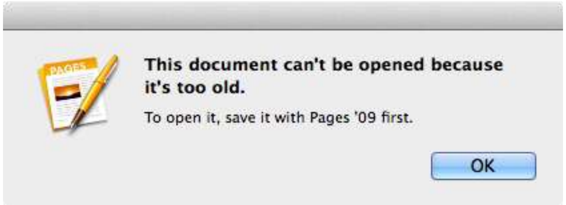
Interface Design



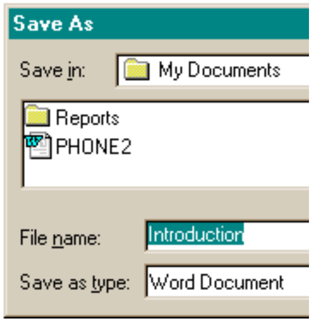
Interface Design



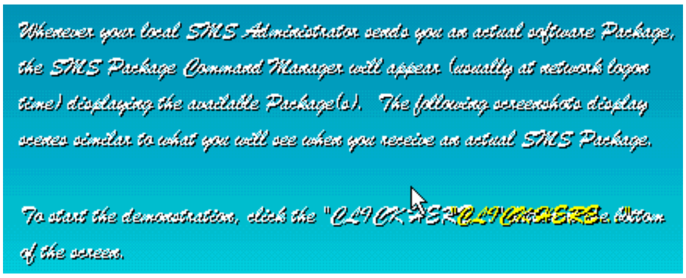
Interface Design



Interface Design



Interface Design



Interface Design

A

B

Accuweather

YahooWeather

Interface Design

Accuweather

Assigment 1

Find out one example of good user interface design and one example of bad user interface design.

Justify your judgements with concrete reasons.

It is difficult to find a perfect interface or a completely bad interface. So, your examples should focus on specific aspects of a user interface (not the whole interface).

The work is not limited to computer software. You can consider any type of interface, such as radios, doors, …

Think of your own everyday experiences.

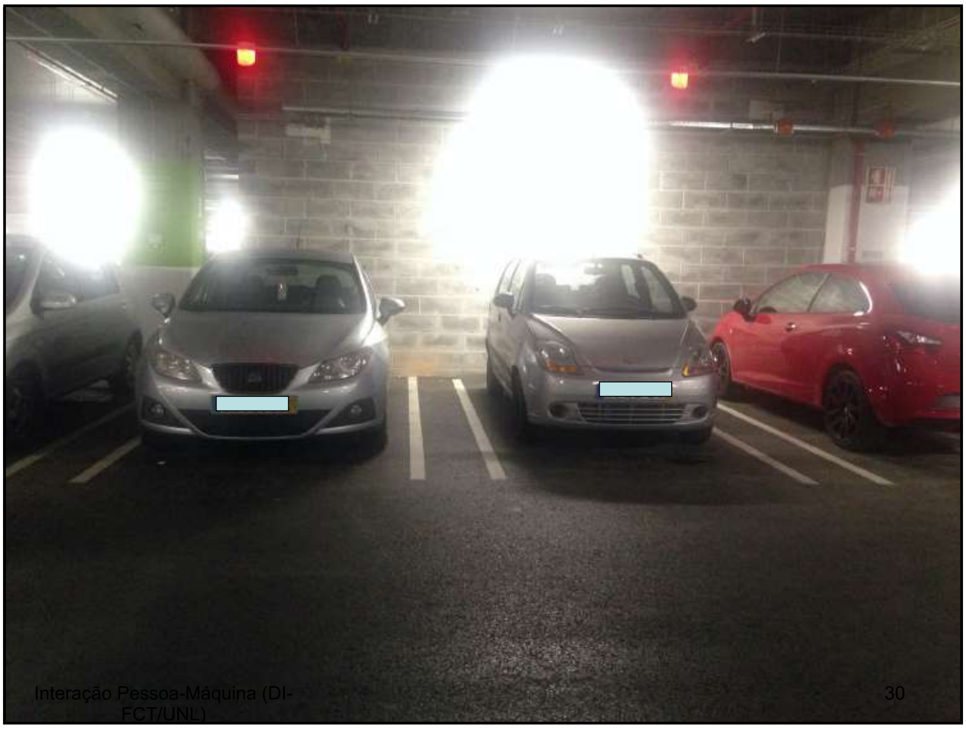
Assigment 1

For each example:

- *describe the objective of the whole interface*
- *describe the good or bad aspects*
- *explain why it is good or bad*
- *for bad cases, tell why, do you think, it was designed that way and suggest corrections or improvements*
- *whenever possible, illustrate with images.*

Focus on minimizing the user's (reader) cognitive effort and maximize his efficiency.

Interação Pessoa-Máquina (DI-FCT/UNL) 29



Interação Pessoa-Máquina (DI-FCT/UNL) 30

Interface Design

Interface Design

Interface Design

What is Human-Computer Interaction (HCI)?

What is HCI?

- Factories in the beginning of XX century
  - human performance evaluation in manual tasks
- Second World War
  - production of more effective weapons
- Formation of the Ergonomics Research Society, 1949.
- Man-Machine Interaction => HCI
- The study of Human-Computer Interaction involves several aspects:
  - Physical
  - Psychological
  - Theoretical

What is HCI?

"HCI involves the design, implementation and evaluation of interactive systems in the context of the user's task."

Dix,2004

What is HCI?

"If I were to sum up interaction design in a sentence, I would say that it's about shaping our everyday life through digital artifacts – for work, for play, and for entertainement."

Gillian Crampton Smith, interview of January 30, 2002

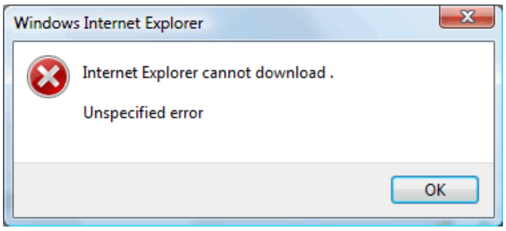
What is HCI?

- Must know:
  - the users
  - the tasks
  - the usage context
- Should apply:
  - Iterative user-centred design
  - Prototyping techniques
  - Usability principles
  - Evaluation techniques

HCI – Why?

- It's not just about "how big should be the buttons?" or "which colour should be used for the background?"
- It can affect:
  - Effectiveness
  - Safety
  - Mood
  - Productivity
  - …

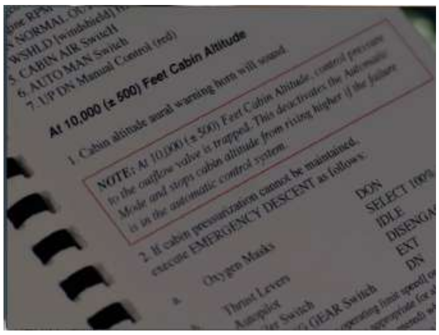
HCI – Why?



Good design: Feedback - error warning
The user knows there is an error

Bad design: Not enough information. The user doesn't know:
- the cause
- the solution

HCI – Why?



Appropriate instructions can make the difference

HCI – Why?

The world's atomic energy authorities have been on alert since reports surfaced linking the **deaths of eight patients undergoing treatment** for pelvic cancer at the National Oncology Institute in Panama City, Panama, to overexposure during radiation therapy.

The International Atomic Energy Association (IAEA), Vienna, Austria, which has been investigating the deaths, said the overdoses probably were not due to a malfunction of the radiotherapy machine, but to a **problem with the system's data entry method**.

In August 2000, the Oncology Institute **changed the process for entering coordinates** for "shielding blocks" designed to protect healthy tissue during radiation therapy. The IAEA report said the change, coupled with a lack of updated written standard procedures, **resulted in miscalculations of radiation intensity and treatment times.**

HCI – Why?

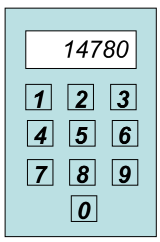
- Users' time isn't getting cheaper
  - Call center with 400 users
  - 750 screens/day
  - 230 days/year
  - User work cost: 5€/hour
  - Reduction of 3s/screen

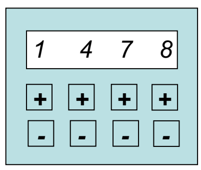
$\Rightarrow$ 287500 € / year

HCI – Why?

HCI – Why?

- Have the user in mind
- Try it out
- Involve the users in the design process
- Iterate





Mechanical syringe controller

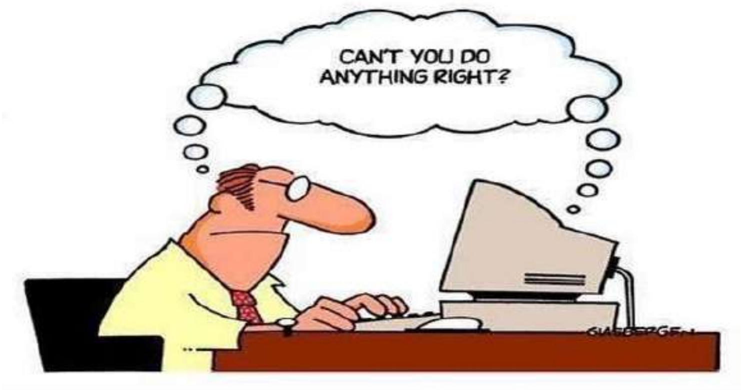
HCI – Why?

– Systems must be robust and consistent.
– Must be prepared for the target users.
– Must support careless usage.
– Should be helpful => help to complete a task instead of creating extra obstacles.
– Interface design shouldn't be handled in the last minute.
  • Interface must be developed along with the rest of the system.

HCI – Related disciplines

• Computer Sciences
• Ergonomics and Human factors
• Artificial Intelligence
• Cognitive Psychology
• Sociology
• Design
• Management
• ...

User Interfaces



User interfaces are hard to design

- **User interface takes a lot of software development effort**

- **UI accounts for ~ 50% of:**
  - Design time
  - Implementation time
  - Maintenance time
  - Code size

(Myers & Rosson, "Survey on user interface programming", CHI '92)

User interfaces are hard to design

- **The user is always right**
  – if users have problems with an aspect of the interface, then there must be something wrong with it
- **The user is not always right**
  – user interface design can not be derived just by asking users what they would like. Users often don't know what is good for them. (ex: Klemmer, *Ergonomics, Ablex, 1989, pp 197-201).*
- **Users are not designers**
  – they don't come up with design ideas from scratch
  – they react to concrete designs they do not like
  – so, we should present suggested designs in a form users can understand (prototypes)

(ex: Grudin & Barnard, "When does an abbreviation become a word?", CHI '85)

User interfaces are hard to design

- **You (the developer) are not a typical user**
  – You are not a domain expert
  – You know far more about your application than any user. It's very hard to forget things you know.
- **You need to communicate with the users**
  – Speak their language.
  – Collect their requirements, communicate your solutions and get their feedback.

Usability

Usability

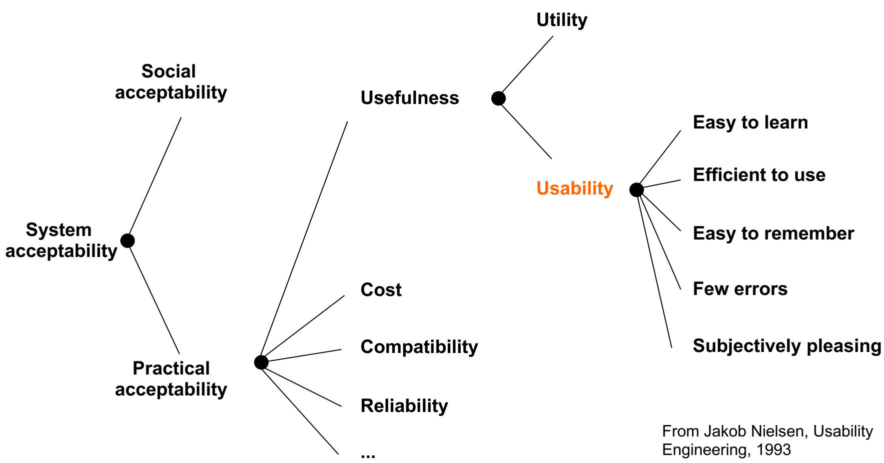
"Knowing some usability principles will help you see the problems yourself and help keep you from creating them in the first place"

Krug, 2014

Usability

The New Yorker, May, 2012

System acceptability



From Jakob Nielsen, Usability Engineering, 1993

Usability definition

- ISO 9241 usability standard

"Effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in a particular environment."

- Functional specifications: crucial to ensuring system functionality
- Usability specifications: crucial to ensuring system usability

ISO 9241 Usability standard

Assume traditional usability principles:

- effectiveness
  - can we achieve what we want to?
- efficiency
  - can we make it without wasting effort?
- satisfaction
  - do we enjoy the process?

Usability

How well users can use the system's functionality?

Usability Attributes

- **Learnability**
  - easy with which new users can begin effective interaction and achieve maximal performance.
- **Efficiency**
  - once the user has learned to use the system, a high level of productivity should be possible.
- **Memorability**
  - should be easy to remember.
- **Errors**
  - should have a low error rate.
- **Satisfaction**
  - should be pleasant to use

(Jakob Nielsen, Usability Engineering, 1993)

Usability measurements

- We can quantify these measures of usability
- Usability is measured relative to certain users (selected to be as representative as possible of the intended users) and certain tasks
- Measurements can be made:
  - in the lab
  - in the wild

Usability - Learnability

- Easy of learn – refers to the novice user's experience on the initial part of the learning curve.
- How do users learn to use a new interface? Most of the time:
  - They don't try to learn it first (there are some exceptions!)
    - They don't read the manual or the online help
    - They don't take a class
  - They try to learn by doing
    - They have a goal
    - Explore the interface to achieve that goal

Usability - Learnability

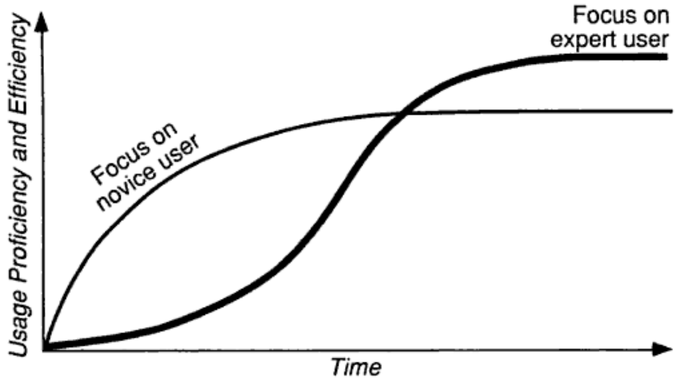
- User interface should clearly communicate how it works and how it is supposed to be used.
- Help (users look for help when they get stuck) should goal-oriented and searchable.
- Highly learnable systems allow users to reach a reasonable level of usage proficiency within a short time.

Usability - Learnability

- Pick some users who have not used the system before and measure the time it takes them to reach a specified level of proficiency in using it.
- Express the specified level of proficiency:
  - state that the users have to be able to complete a certain task successfully.
  - specify that users need to be able to complete a set of tasks in a certain minimum time before one will consider them as having "learned" the system.

Usability - Learnability

- Learning curve



From Jakob Nielsen, Usability Engineering, 1993

Usability – Efficiency of use

- Efficiency refers to the expert user's steady state level of performance at the time when the learning curve flattens out.

- Users are considered experienced (expert users):
  - if they say so
  - if they have been users for more than a certain amount of time.

Usability – Efficiency of use

- Experience can also be defined by the number of hours spent using the system.
  - test users are asked to use the system for a certain number of hours, after which their efficiency is measured.

- Continuously measure user's performance (ex: in terms of number of seconds to do a specific task) until it stops to increase, when the user is considered to reach the steady-state level of performance.

Usability – Memorability

- Casual users are people who use a system intermittently (expert users use the system frequently).

- In contrast to novice users, casual users have used the system before and do not need to learn it all from scratch.

- Casual users only need to remember how to use the system based on their previous learning.

Usability – Memorability

- Standard user test with casual users that have been away from the system for a certain time.
- Memory test: after users finish a test session, ask them to explain the effect of various commands or the name of a command that does a certain thing (assess the number of correct answers).
  - GUI – Recognition vs Recall
  - Mayes et al., 1988

Usability – Errors

- Users should make as few errors as possible when using a computer system.
- Error: action that does not accomplish the desired goal.
  - Norman's mistakes and slips
- Error rate is measured by counting the number of such actions made by the user while performing a certain task.
  - Can be measured simultaneously with other usability attributes

Usability – Errors

- Some errors are immediately corrected by the user and have no other effect than to slowdown the user's task completion rate.
  - Need not to be counted separately, as their effect is included in the efficiency of use.
- Catastrophic errors should be counted separately from minor errors and special effort should be made to minimize their occurrence and frequency.

Usability – Satisfaction

- How pleasant it is to use the system.
- Psychophysiological measures (pupil dilatation, blood pressure, heart rate):
  - often intrusive
- Simply ask the users for their subjective opinion (average of multiple answers).
- The most difficult episode a user experience is the most memorable one.

Usability – Satisfaction

- Questionnaires:

Users are asked to rate the system on 1-5 or 1-7 rating scales that are normally either:

  - **Likert scale** – users indicate their level of agreement with certain statements.

    "It was very easy to learn how to use the system."

    Strongly Disagree 1 2 3 4 5 Strongly Agree

  - **Semantic differential scale** - lists two opposite terms along some dimension and asks the user to place the system on the most appropriate rating along the dimension.

    Please mark the positions that best reflect your impressions of the system:

Pleasing	_ _ _ _ _	*Irritating*
Complete	_ _ _ _ _	*Incomplete*
Fast to use	_ _ _ _ _	*Slow to use*

Usability – Satisfaction

- Questionnaires:
  - No matter what rating scales are used, questionnaires should be subjected to **pilot testing** to make sure that the questions are interpreted properly by the users.
  - Users tend to be positive, unless they have had a really unpleasant experience. This phenomenon can be partly counteracted by using **reverse polarity** on some questions.
  - Final rating for subjective satisfaction is often calculated as a mean of the ratings for the individual answers (after compensating for any use of reverse polarity).
  - If multiple systems are tested, subjective satisfaction can be measured by asking users which system they prefer and how strongly they prefer various systems over the others.

Usability – Satisfaction

- Questionnaires:

  - SUS – System Usability Scale
  - USE - Usefulness, Satisfaction, and Ease of use
  - QUIS - Questionnaire for User Interface Satisfaction
  - UEQ – User Experience Questionnaire

Usability – Satisfaction

- SUS – System Usability Scale:

1. I think that I would like to use this system frequently
2. I found the system unnecessarily complex
3. I thought the system was easy to use
4. I think that I would need the support of a technical person to be able to use this system
5. I found the various functions in this system were well integrated
6. I thought there was too much inconsistency in this system
7. I would imagine that most people would learn to use this system very quickly
8. I found the system very cumbersome to use
9. I felt very confident using the system
10. I needed to learn a lot of things before I could get going with this system

Calculate overall score. John Brooke, J. (1996)

Usability – Trade-offs

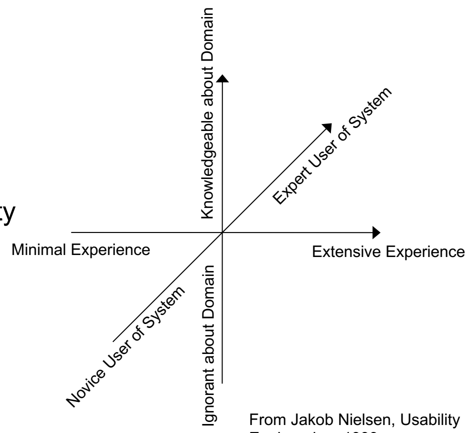
- Not all usability aspects can be given equal weight in a given design project.
- It is not always possible to achieve optimal scores for all usability attributes simultaneously.
  - avoiding catastrophic errors may lead to a user interface that is less efficient to use.
- When usability trade-offs seem necessary, try to find a win-win solution that can satisfy both requirements.
- If that is not possible, define which usability attributes are the most important given the specific circumstances of the project (user & task analysis).

Usability – Trade-offs

- Depends on the user

  Typically:
  - Novices – need learnability
  - Expert – need efficiency
  - Infrequent – need memorability

- Depends on the application



From Jakob Nielsen, Usability Engineering, 1993

Usability – Trade-offs

- Considerations other than usability may lead to designs violating some usability principle.
  - ex: security considerations often require access controls that are non-user friendly – error message in login.
- Make priorities clear on the basis of users and task analysis
  - ex:
    - learnability – when new employees are constantly being brought in on a temporary base
    - memorability – when application is used periodically, once every 3 months.

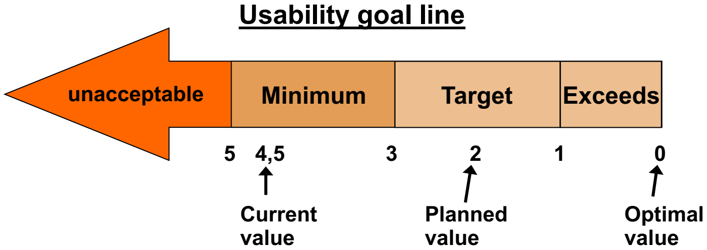
Usability – Trade-offs

- "best of both worlds"
  - accelerators – user interface elements that allow the users to perform frequent tasks quickly, even though the same task can also be performed in a more general, and possibly slower way. Ex: function keys, command name abbreviations, ...

Usability – Goal setting

- For each usability attribute of interest, several different levels of performance can be specified as part of the goal-setting process.

Usability goal line



From Jakob Nielsen, Usability Engineering, 1993

Usability – goal setting

- Usability goals are reasonable easy to set for new versions of existing systems or for systems that have a clearly defined competitor on the market
  - Minimal acceptable usability = current usability level
  - Target usability = sufficiently large improvement to induce changes on the system
- For complete new systems without any competition, usability goals are much harder to set
  - Define a set of sample tasks and ask several usability specialists
  - Get an idea of the minimum acceptable level by asking users (could be dangerous!)

Usability – Good design

"Every designer wants to build a high-quality interactive system that is admired by colleagues, celebrated by users, circulated widely, and imitated frequently."

(Shneiderman, 1992)

User Experience (UX) Design






User Experience (UX) Design

Wake-up experience

User Experience (UX) Design

Waking up with fresh orange juice

- *Different technology*
- *Different user interface*
- *Comparable usability*
- *Same outcome*

So quiet!
Such a different experience!

Buxton, 2005

User Experience (UX) Design

Waking up with fresh orange juice

- *Similar look*
- *Same user interface*
- *Same outcome*

- … incomparable experience

Buxton, 2005

User Experience (UX) Design

- *Similar look*
- *Same user interface*
- *Same outcome*

- *Different technology*
- *Different user interface*
- *Comparable usability*
- *Same outcome*

So quiet!

Such a different experien

- … incomparable experience

Buxton, 2005

User Experience (UX) Design

Buxton, 2005

Usability

"Let's design systems to fit people instead of the other way around."

Randolph Bias

Additional reading

Buxton, B. Experience Design vs. Interface Design. Rotman Magazine, pp. 47-49, 2005.

http://www.billbuxton.com/experienceDesign.pdf

References

- Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & A. L. McClelland (Eds.), Usability Evaluation in Industry. London: Taylor and Francis.
- Buxton, B. Experience Design vs. Interface Design. Rotman Magazine, pp. 47-49, 2005. http://www.billbuxton.com/experienceDesign.pdf
- Dix, Alan, Finlay, Janet, Abowd, Gregory, Beale, Russel. Human-Computer Interaction (3rd Edition), Prentice Hall, 2004.
- Krug, S., Don't make me think, Revisited: A common Sense Approach to web Usability. New Riders, 2014.
- Nielsen, Jakob, *Usability Engineering*, Academic Press, 1993.
- Shneiderman, Ben. Designing the User Interface: Strategies for Effective Human-Computer Interaction (third edition). Addison-Wesley, Reading, Massachusetts, 1998.
- Mayes, J.T., Draper, S.W, Mcgregor, A.M. and Oatley, K. Information Flow in a User Interface: The Effect of Experience and Context on the Recall of MacWrite Screens. In Jones, D.M. and Winder, R. (Eds.), People and Computers IV. Cambridge University Press, Cambridge, UK, pp. 275-289,1988.
- Myers, B. and Rosson, M. B., Survey on user interface programming. Proc. of CHI '92, pp. 195-202.

Interação Pessoa-Máquina

Teresa Romão
tir_at_fct.unl.pt
DI/FCT/UNL

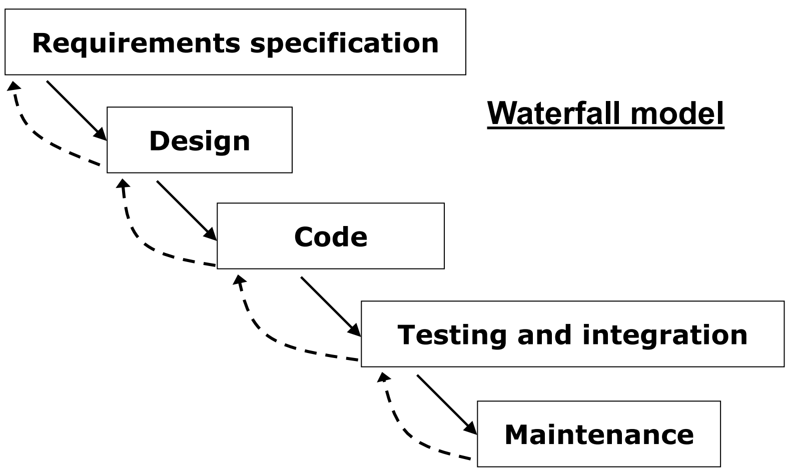
2022/2023

The Design Process

Software life cycle

- 60´s e 70´s – software crisis
  - **Code now, fix it later...** – process needs some structure
- Waterfall model – attempt to organize the development process – "**think first and code second**".

Software life cycle



Software life cycle

- Requirements specification
  - designer and customer try to establish **what** should be implemented.
  - involves collecting information concerning the users and the working environment or domain in which the final product will function.
  - working domain aspects include:
    - function that the software should perform
    - details about the environment in which it must operate
    - people it will potentially affect
    - relationship with other existing products which it is updating or replacing.
  - Use a language that both users and developers understand.

User and task analysis

Software life cycle

- Design
  - Establish how the system should be implemented, in order to fulfil the requirements specification.
  - Produce an appropriate detailed description of interface design, in order to be implemented in a programming language.
- Code
  - implementation and testing of the individual modules in a programming language.
- Testing and integration
  - Includes further testing and acceptance tests with users to ensure the system meets the requirements.

Software life cycle

- Maintenance
  - after the system release, maintenance lasts until a complete redesigned new version is produced or the system is phase out completely.
  - comprises:
    - correction of errors discovered after release
    - system review to satisfy requirements that have not been identified earlier
    - ... so, there is feedback between this activity and all others.

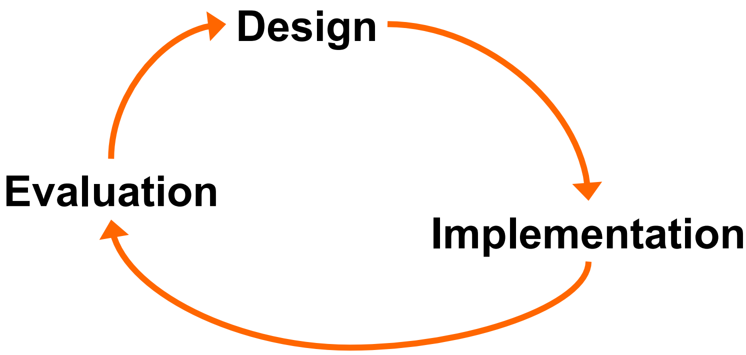
Waterfall model

- Waterfall model is not appropriate for interactive systems interface design:
  - users only participate in the requirements specification and testing.
  - Late detection of errors causes expensive and long lasting rectifications.
  - no support for really iterative processes.

Waterfall Model

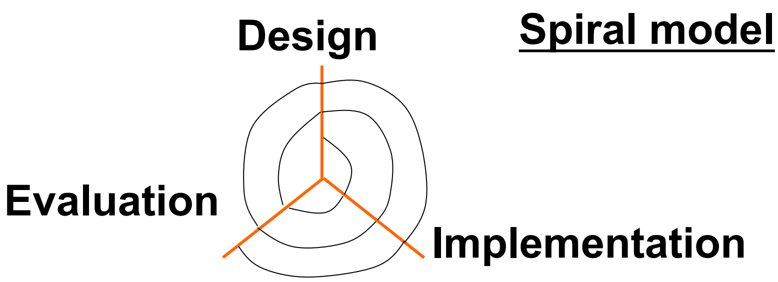
- Wrong: follow the waterfall model, produce a bad interface and release the system.
  - Each iteration corresponds to a version of the system
    - Errors detected during evaluation are corrected in the next version.
  - Clients are used to evaluate the system's usability:
    - If they don't like, they don't buy the next version!

Iterative design



worst case in the waterfall model?

Iterative design



Iterative design

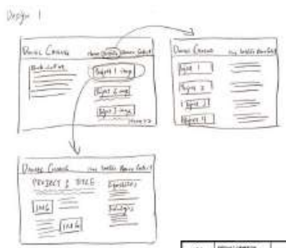
- Spiral model
  - Several iteration
    - Cost, accuracy and correctness increase in each iteration.
  - First iteration may be done in paper: low cost, ...and far from what it will look like.

Iterative design

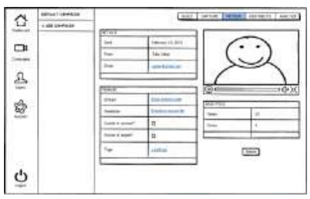
- First iterations – low cost prototypes
  - Parallel design: development and testing of several prototypes to explore multiple alternatives.
- Subsequent iterations (after eliminating the highest risks)
  - creation of more elaborated prototypes
- Every prototype is evaluated
  - users are involved in every iteration
- More iterations → better interfaces
- Only better interfaces survive and reach the market.

Iterative design

Sketches



Paper prototype



Computer Mockup

User and task analysis

User-Centered Design

- Science finds, Industry applies, Man conforms - Slogan of World Fair Chicago 1933.
- People propose, Science studies, Technology conforms - Slogan de Donald Norman.

User-Centered Design

- The design is based upon users':
  - needs,
  - abilities,
  - context,
  - work,
  - task.

User and task analysis

User-centered design – first steps

- User analysis: Who are the users?
- Task analysis: What does the user need to do?

User and task analysis

- Collect data about the users and tasks (characteristics and needs) ...
- ... represent the data, in order to make interpretation easier and guide the design.

User and task analysis

	User and task analysis	**System analysis**
Main focus	People (users)	Computer
Goal	Collect data for UI design	Collect data for the software and data structure design
Results	List of tasks and users' profiles	Functional specification and architecture

Based on Gonçalves, 2017

Know the user

- Identify characteristics of the target user population:
  - age, gender, ethnicity
  - Education
  - Physical abilities
  - General computer experience
  - Domain experience
  - Application experience
  - Work environment and social context
  - Communication patterns

Know the user

- Many applications have several kinds of users.
- Need to analyse all kinds of users.
- Example: Self-service supermarket checkout
  - Clients
  - Assistants
- Describe what the real users are rather than what you want them to be.

Know the user

- Techniques
  - Questionnaires
  - Interview
  - Observation
- Obstacles
  - Some users are hard to reach
  - Users speak another language

Example: Self-service supermarket checkout

- Who are the users?
  - Shoppers
  - Age: 10-80 years
  - Different physical abilities: height, strength,...
  - No training: arrive and use
  - Knowledge of food, but not about supermarket stock management.
  - Can ask each other for help.
- Main user classes:
  - Clients
  - Assistants

Users description

Table 5 ATM user groups (adapted from Stone, 2001)

User characteristic	ATM customer characteristics, by group		
	Teens/Young adults	*Young adults to middle age*	*Middle age to senior citizens*
Age	12 to 25.	25 to 50.	50 to 80+.
Sex	Both male and female.	Both male and female.	Both male and female.
Physical limitations	May be fully able-bodied, or may have some physical limitations in relation to, for example, hearing or sight. Will be of varying heights.	May be fully able-bodied, or may have some physical limitations in relation to, for example, hearing or sight. Will be of varying heights.	May be fully able-bodied, or may have some physical limitations in relation to, for example, hearing or sight; mobility, or use of hands. Will be of varying heights.
Educational background	May have minimal or no educational qualifications.	May have only minimal educational qualifications.	May have only minimal educational qualifications.
Computer/IT use	Probably have some prior experience of computer or IT use.	May have little or no prior experience of computer or IT use.	May have little or no prior experience of computer or IT use.
Motivation	Probably very motivated to use the ATM, especially in relation to their banking habits.	Could be very motivated to use the ATM, especially if they can do their banking quickly and avoid queuing in a bank.	Could be very motivated to use the ATM, but would probably prefer to stand in a queue in the bank.
Attitude	Attitudes to use may vary, depending on the services the automated teller offers and the reliability of the technology itself.	Attitudes to use may vary, depending on the services the automated teller offers and the reliability of the technology itself.	Attitudes to use may vary, depending on the services the automated teller offers and the reliability of the technology itself.

Task analysis

- Identify the users' goals and study the way users perform their jobs.
  - What users do?
  - Why they do it?
  - How they do it?
  - What they must know?
  - What tools they use?
- The new system/interface may change the current processing ("How?")
- Understanding "how" and "why" allows for a deeper knowledge about the tasks.

Task analysis

- Study of the way people perform tasks with existing systems.

- Users can help you learn:
  - What is related to their job performance
  - What instruments do they use
  - What they actually do

- You show the possibilities of technology
  - Build a relationship and convey an idea of what is possible
  - Users can comment if ideas make sense

Task analysis

- The general method for Task Analysis is:

  - Observe / Ask
  - collect unstructured lists of words and actions
  - organize using notation or diagrams

Task analysis

- Identify the individual tasks the system should perform
- Each task represent a goal (what?, not how?)
- Top-down approach: start with the overall goal of the system and decompose it hierarchically into tasks
- Overall goal: self-service checkout
  - Tasks:
    - Register products
    - Pack
    - Pay

Task analysis

- What needs to be done?
  - Goal
- What must be done to make it possible?
  - Pre-conditions
    - Tasks on which this task depends
    - Information the user needs to know
- What steps are involved in doing the task?
  - Sub-tasks
  - Sub-tasks may be decomposed recursively.

Example: Self-service supermarket checkout

- Goal
  - Register products
- Pre-conditions
  - All the desired products are in the cart
- Sub-tasks
  - Register pre-packaged product
  - Register loose product

Example: Self-service supermarket checkout

- Where is the task performed?
  - Supermarket exit, standing up
- How often is the task performed?
  - once a week
- What are its time or resource constraints?
  - 3 minutes
- How is the task learned?
  - Try it
  - Watching others
  - Assistant demo
- What can go wrong? (exceptions, errors, emergencies)
  - Bar code is missing or unreadable
- Who else is involved in the task?

Task analysis

- Collecting information techniques:
  - Direct observation of users performing tasks
  - Interviews with users
  - Contextual inquiry
  - Participatory design
  - Expert advice
  - Documentation analysis
  - Logging

Example: Self-service supermarket checkout

- Observe store cashiers checking out products to understand the supermarket checkout task.

- Interview shoppers to better understand their goals.

Observation

- Real environment (animals in a zoo) versus controlled environment (video).
- Passive (watch and hear - record) versus active (ask)
- Encourage the user to think aloud
- Capture what the users say and do
- Describe the observation to someone who have never witnessed the task

Observation

- Questions to ask:
  - Why do you do this? (goal)
  - How do you do this? (sub-task)
  - What must be done before doing this? (sequence, pre-conditions)
  - What fails when you do this?(exceptions)
- Look for the weaknesses in the current system
  - Goals not accomplished, wasted time, user irritation
- At the end: “What else should I ask?”

Observation

- Dangers (direct observation):
  - Duplicate bad existing procedures
  - Failing to capture good existing procedures

  Know: Why users do what they do (not just what they do!)

Interviews with the users

- **Structured**
  - Follow an interview plan
  - Be specific
  - Efficient
  - Needs preparation
- **Non structured**
  - Open talk
  - Inefficient
- **Semi-structured**
  - Start with a plan of questions and end up in an open talk
  - Balanced
  - Often appropriate
- Record interviews (when appropriate and with consent)

Interviews with the users

- **Plan your questions:**
  - How do you perform task X?
  - Why do you perform task X?
  - When (what conditions) do you perform task X?
  - What do you do before you perform task x?
  - What information do you need for…?
  - Who are the persons you need to communicate for …?
  - What do you use for…?
  - What happen after performing…?
  - What is the result of…?
  - What are the consequences of not doing…?

Contextual inquiry

- Contextual inquiry
  - Combines interviewing and observation in the user's actual work environment, discussing actual work products.
  - Fosters strong collaboration between the designers and the user.
  - Be concrete
  - Establish a master-apprentice relationship
    - User shows how and explains
    - Interviewer watches and asks questions.

Participatory design

- Instead of guessing, designers should have access to a pool of representative users.
- Include representative users directly in the design team.

Participatory design

- Periodical refresh of the pool of users who participate in large projects
  - users become less representative as they understand the proposed system structure
- Changing users representative involves spending time explaining the project to new users.

Expert advise

- Experts describe tasks as they should be executed ...
- ...not necessary, how they are actually executed.

Documentation analysis

- Describe how it should be done...
- ...instead of how it is done.
- Try to understand why it is not done “by the book”.

Logging

- Keystrokes / mouse clicks
- Transactions logs
- Location
  - Mobile phones
  - RFID
  - GPS

Study concurrent products

- Search for good and bad ideas:
  - Funcionalities
  - Interaction styles

Task description

- After collecting the information, we need to organize and represent it in order to guide the design

- Even the simplest task may become quite complex to describe

  – For example, sending an email.
  Easy, right?

Task description

- Sending an email:
  – Click New Email button
  – Click inside the “to:” field
  – Type recipient’s email address
  – Click inside the subject field
  – Type the subject of the email
  – Click inside the body field
  – Type email, including a greeting and closing sentence.
  – Add signature
  – Double-check email for correct spelling and grammar
  – Click Send button

Task description

1. Identify the task to analyse
2. Break down the task into subtasks
3. Identify steps in subtasks

Task description

- Sending an email:
  - Click New Email button
  - Enter recipient
    - Click inside the "to:" field
    - Type recipient's email address
  - Define the subject
    - Click inside the subject field
    - Type the subject of the email
  - Write message
    - Click inside the body field
    - Type email, including a greeting and closing sentence.
    - Add signature
  - Double-check email for correct spelling and grammar
  - Click Send button

Task analysis

- "Hierarchical task analysis"- HTA
  - Hierarchical decomposition of tasks
  - Specification plan describing in what order and under what conditions subtasks are performed.
  - Start point: user goal.

Task analysis

- Hierarchical task analysis (HTA) can be textually or graphically represented.

Hierachical task analysis (HTA)

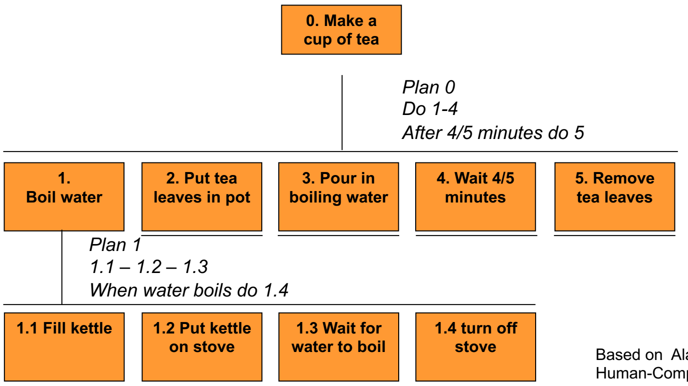
0. Make a cup of tea
- **1.** Boil water
  - **1.1** Fill kettle
  - **1.2** Put kettle on stove
  - **1.3** Wait for water to boil
  - **1.4** Turn off stove
- **2.** Put tea leaves in pot
- **3.** Pour in boiling water
- **4.** Wait 4/5 minutes
- **5.** Remove tea leaves

Plan 0
Do 1- 4
After 4/5 minutes do 5

Plan 1
1.1 – 1.2 – 1.3
When water boils do 1.4

Task analysis

- Hierarchical task analysis (HTA) can be textually or graphically represented.



Based on Alain Dix et. al, Human-Computer Interaction, 2004

Task analysis

- After the first approach to the task's description: verify errors and omissions.
- Possible approach: consult an expert.
  - Omission: warm pot.
- Examine sub-tasks
  - 1.4 turn off stove. When was it turn on? Implicitly in 1.2.
- Balance the hierarchy (may not be necessary or desirable!)

Task analysis

0. Make a cup of tea
- **1.** Boil water
  - **1.1** Fill kettle
  - **1.2** Put kettle on stove
  - **1.3 Turn on stove**
  - **1.4** Wait for water to boil
  - **1.5** Turn off stove
- **2. Make pot**
  - **2.1 Warm pot**
  - **2.2** Put tea leaves in pot
- **3** Pour in boiling water
- **4.** Wait 4/5 minutes
- **5.** Remove tea leaves

Plan 0

Do 1- 3
At the same time 2
Then 3-4
After 4/5 minutes do 5

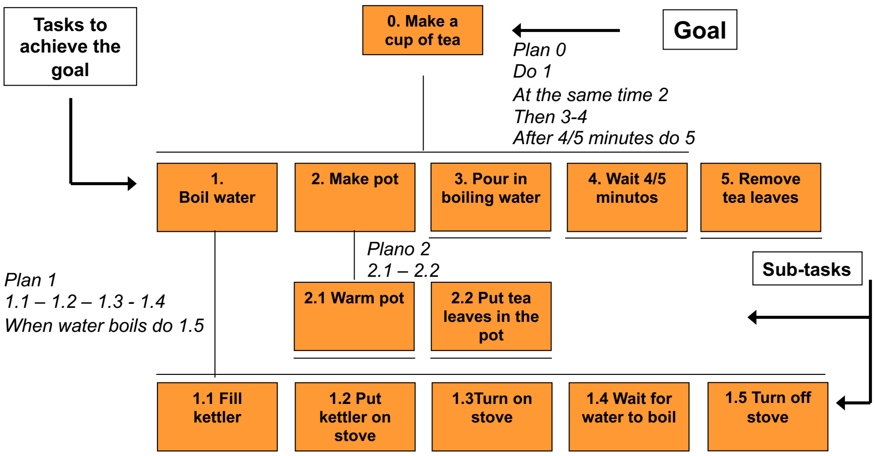
Plan 1

1.1 – 1.2 – 1.3 –1.4
When water boils do 1.5

Plan 2

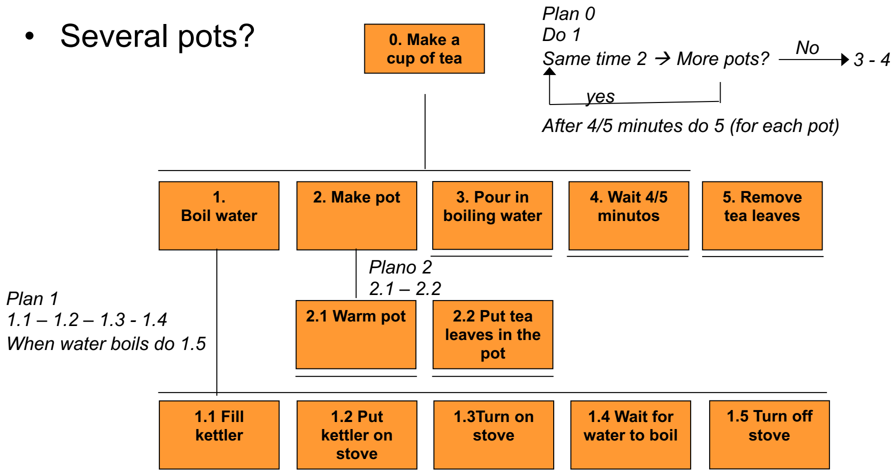
2.1 – 2.2 – 2.3

Task analysis



Task analysis

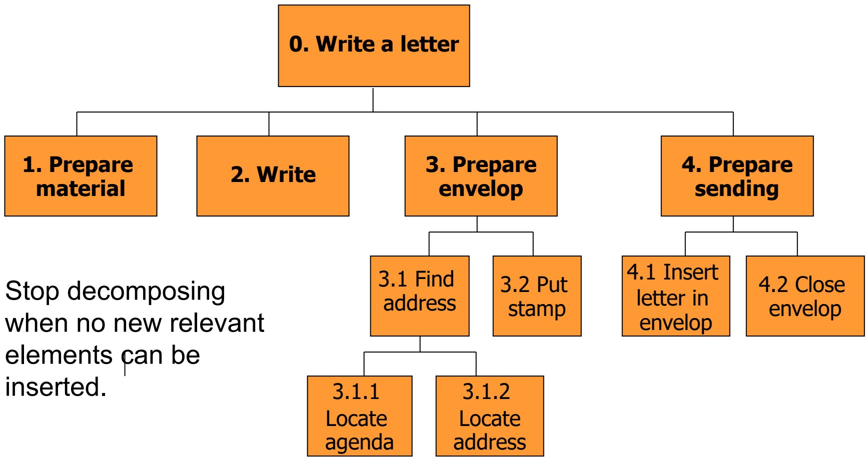
- Several pots?



Task analysis

- Types of tasks
  - Fixed sequence (plan 2)
  - Optional tasks (add sugar as task 6)
  - Waiting for events (plan 0 e 1)
  - Cycles (plan 0)
  - Time-sharing (task 1 and 2 can be done at the same time)

Task analysis



Stop decomposing when no new relevant elements can be inserted.

Task analysis

- User X
  - Gather paper, pen, envelop and stamp
  - Write letter
  - Fill in envelop
  - Stick the stamp
  - Insert letter in envelop
  - Close envelop
- User Y
  - Gather paper, pen, envelop and stamp
  - Fill in envelop
  - Write letter
  - Insert letter in envelope
  - Stick the stamp
  - Close envelop
- And user Z?

Task scenarios

- Based on narratives that describe:
  - Actors
  - Objectives
  - Tools
  - Thoughts/Actions/events (sequence) to achieve the goals

Task scenarios

- Informal description narrative
- Uses the user vocabulary
- Repetitive references to an object or behaviour may suggests its importance or relevance in the context.
- Scenarios describing the actual situation may help to define new scenarios.
- Provide test cases

Task scenarios

- Task scenario describing the use of a library catalogue:
  - "Say I want to find a book by George Jeffries. I don't remember the title but I know it was published before 1995. I go to the catalog and enter my user password. I don't understand why I have to do this, since I can't get into the library to use the catalog without passing through security gates. However, once my password has been confirmed, I am given a choice of searching by author or by date, but not the combination of author and date. I tend to choose the author option because the date search usually identifies too many entries. After about 30 seconds the catalog returns saying that there are no entries for George Jeffries and showing me the list of entries closest to the one I've sought. When I see the list, I realize that in fact I got the author's first name wrong and it's Gregory, not George. I choose the entry I want and the system displays the location to tell me where to find the book."

From Interaction Design: Beyond Human-Computer Interaction, 2nd Edition; Sharp, Rogers, Preece. 2007.

Task scenarios

Caller:	(Dials 233-888-8888.)
Operator:	Irish National Olympic Committee. Can I help you?
Caller:	I want to leave a message for my son, Michael.
Operator:	Is he from Ireland?
Caller:	Yes.
Operator:	How do you spell his name?
Caller:	K-E-L-L-Y
Operator:	Thank you. Please hold for about 30 seconds while I connect you to the Olympic Message System.
Operator:	Are you ready?
Caller:	Yes.
OMS:	When you have completed your message, hang up and it will be automatically Sent to Michael Kelly. Begin talking when you are ready.
Caller:	"Michael, your Mother and I will be hoping you win. Good luck." (Caller hangs up.).

Example of a Parent Leaving
a Voice Message for an Olympian (from Gould, 1987)

Task scenarios for usability testing

- A narrative that describes the action that you ask the participant to take on the tested interface.
- Need to **provide context** so users engage with the interface and pretend to perform the tasks as if they were at home or in the office.

- Example:

  You're planning a vacation to Katmandu, April 10 – April 24. You need to buy both flights and hotel. Use application X to find the best deals.

Task scenarios for usability testing

- Do not give clues nor describe the steps
- Avoid terms used in the interface

- Example:

  User goal: Check grades.

  Poor task scenario: You want to check the results of your exams. Go to the website X, sign in, click on Courses -> Grades.

  Better task scenario: Look up the results of your exams in website X.

Task scenarios for usability testing

- Avoiding clues does not mean being vague
- Example:

Poor task scenario: Make an appointment with your doctor.

Better task scenario: Make an appointment for next Wednesday at 4pm with your doctor, Dr. Philips.

Aplications (task analysis)

- User guide production
- Requirements specification
- Detailed interface design
  - Lists of object/action suggest interface elements
  - Sequences of actions guide the dialog design.

Example: IDEO

https://www.youtube.com/watch?v=GYkb6vfKMI4

Example: IDEO

https://www.youtube.com/watch?v=M66ZU2PCIcM

Generating ideas

- First, think about and write down your individual ideas
- Then, brainstorm: come together to discuss and build upon each other's ideas.
- Get everything on the board

Generating ideas

- Some [IDEO tips on better brainstorming](https://www.ideou.com/pages/brainstorming): (https://www.ideou.com/pages/brainstorming)
  1. Defer judgment
  2. Encourage wild ideas
  3. Build on ideas of others
  4. Stay focused on the topic
  5. One conversation at a time
  6. Be visual
  7. Go for quantity

Summary

- Define the necessary data
- Collect data using the different methods and techniques
- Represent tasks and sub-tasks
- Use these data as the basis for design

- Be efficient!

References

- Dix, Alan, Finlay, Janet, Abowd, Gregory, Beale, Russel. Human-Computer Interaction (3rd Edition), Prentice Hall, 2004.

- Gonçalves, D., Fonseca, M.J., and Campos, P., Introdução ao Design de Interfaces. FCA, 2017.

- Preece, Rogers and Sharp, Interaction Design, Wiley, 2002.

Assignment : Read and analyse

- John Gould et al., The 1984 Olympic Message System: a test of behavioral principles of system design. *Communications of ACM*, v.30 n.9, 1987. (http:/doi.acm.org10.1145/30401.30402)
  You tube video: https://www.youtube.com/watch?v=W6UYpXc4czM&feature=related

Interação Pessoa-Máquina

Teresa Romão
tir_at_fct.unl.pt
DI/FCT/UNL

2022/2023

Evaluation - Dates

Tests (will be on Clip when confirmed):
- T1: November 7, 19h
- T2: December 13, 19h

Mandatory lab classes:
- Prototype testing day: October 18, 19 and 20
- Heuristic evaluation
- Project presentation

Testing Day

October 18/19/20

Sketching

The role of design

- Explicit design process
  - Preproduction in film making
  - Development of a new automobile
- Need to insert a design process at the front end of product development
  - The cost and time lost due to this additional stage will be significantly less than the cost and time lost due to the poor planning and overruns that will result if it is not included.
- Dangerous assumptions:
  - We know what we want at the start of a project
  - We know enough to start building it

The role of design

- Problem setting
  - What is the right thing to build?
- Problem solving
  - How do we build this?

- You must **get the right design** as well as **the design right**.

Sketching

- Communicate ideas
- Aid of thought

Sketching

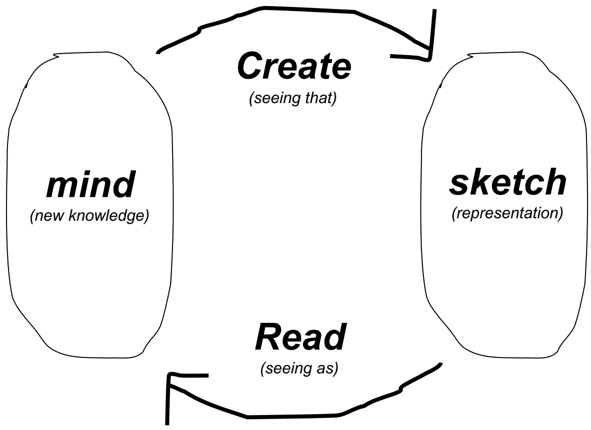
Attributes of sketches:

- Quickly / Timely
- Inexpensive / Disposable
- Plentiful
- Clear vocabulary
- Minimal detail
- Appropriate degree of refinement
- Suggests and explore rather than confirm
- Ambiguity

Sketching

- To get the most out of a sketch, we need to leave big enough holes
- Ambiguity creates the holes
- It enables a sketch to be interpreted in different ways, even by the person who created it.

Sketching



From Sketching user experiences, Buxton, 2007

Sketching

"…designers do not draw sketches to externally represent ideas that are already consolidated in their minds. Rather, they draw sketches to **try out ideas**, usually vague and uncertain ones. By examining the externalizations, designers can **spot problems they may not have anticipated**. More than that, they can see new features and relations among elements that they have drawn, ones not intended in the original sketch. These unintended discoveries **promote new ideas and refine current ones**. This process is iterative as design progresses."

Suwa and Tversky, 2002

Sketching

"The best way to a good idea is to have lots of ideas"

Linus Pauling

- Exercising the imagination and understanding (mental and experiential)
- Chose appropriate materials

Sketches and prototypes

<u>*Ceramic class*</u>

Group 1 – Quantity

How many pots of which level of quality?

Group 2 – Quality

One pot… one perfect pot (?)

Who produced the highest quality work?

From Bayles and Orland, 2001

Sketching

IF	Sketching is Fundamental to Design
AND	We are Designing Interactive Systems
THEN	How do you sketch interaction?
	What are the fundamental skills?
	What is the fundamental process?

Sketching in interaction design can be though of analogous to traditional sketching. Sketches need to be able to capture the essence of design concepts around transitions, dynamics, feel, ...

Sketching

Sketching in Interaction Design

- Analogous to traditional sketching
- Shares all of the same key attributes
- More feel than look
- Must accommodate time & dynamics

Sketches and prototypes

Sketching → ***Prototyping***

Time

Suggest → *Describe*
Explore → *Refine*
Question → *Answer*
Propose → *Test*
Provoke → *Resolve*

Sketches and prototypes

Sketching ***Prototyping***

→ ***Time***

Low investment
More opportunities to explore

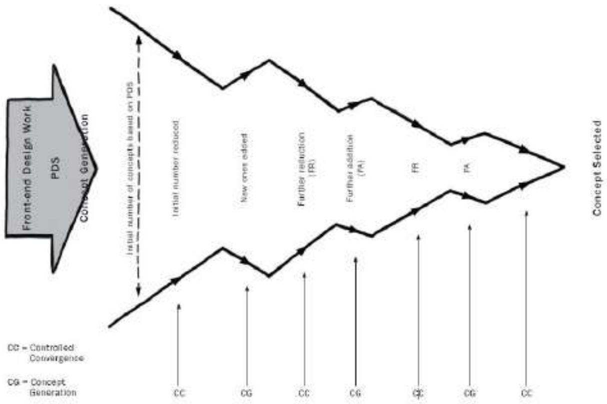
Fail early … and learn

Sketches and prototypes

- Objective:
  - design products that people want, need, like and can use.
- Mean
  - earlier iterative user involvement
    - → user input should begin early enough to influence the design

Sketches and prototypes

Controlled convergence



Pugh, 1990

Prototyping

Prototyping

- The requirements of an interactive system can not be completely specified on the first SLC activity.
- To be sure about some potential design functionalities, you have to build and test them with real users.
- The design can then be modified in order to correct some false assumptions revealed during the tests.
- Iterative design:
  - "a purposeful design process which tries to overcome the inherent problems of incomplete requirements specification by cycling through several designs, incrementally improving upon the final product with each pass."

Human-Computer Interaction, Alan Dix et al., 1998

Prototyping

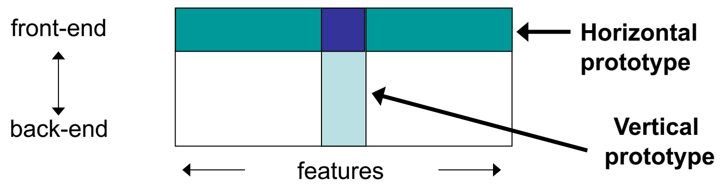
- Why?
  - Faster development, earlier feedback
  - Cheap
  - Makes parallel design easier
  - Easy to modify and throw away
  - The activity of building prototypes encourages reflection in design
  - User-centered design
    - "Experience shows that it is not possible to involve the users in the design process by showing them abstract specification documents, since they do not understand them nearly as well as concrete prototypes".

From Jakob Nielsen, Usability Engineering, 1993

Prototyping

- Fidelity
  - Low: omits details, uses cheap materials.
  - High: more like the final product.

  Dimensions
  - Breadth: Number of features in the prototype
    - Only enough features for certain tasks
  - Depth: Implementation degree of each feature
    - conditioned responses, no error handling



Prototyping

- Fidelity

  Dimensions
  - "Look": Appearance, graphic design
  - "Feel": Sensation, physical method of interaction
    - Point and write ≠ mouse and keyboard

Prototyping

- Non-computational vs computational

Early stages of development

Final stages of development

Low fidelity

High fidelity

Prototyping

- Prototypes can be produced faster by:
  – placing less emphasis on the efficiency of the implementation
  – accepting less reliable or poor quality code
  – using simplified algorithms
  – wizard of Oz approach
  – using low-fidelity media
  – using fake data and other content
  – using paper mock-ups instead of a running computer system

Jakob Nielsen, 1993.

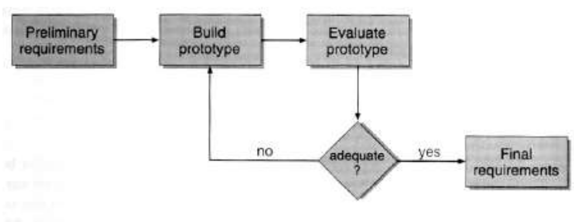
Prototyping

- 3 approaches to prototyping:
  - "Throw-away"
  - Incremental
  - Evolutionary

Dix, Alan, et al., 1998.

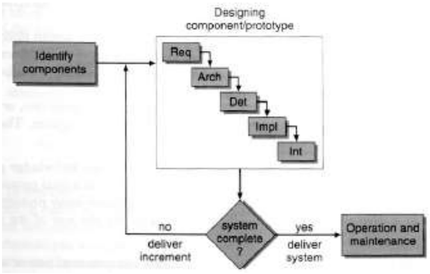
Prototyping

- "Throw-away":
  - The prototype is built and tested. The knowledge gained by this exercise is used to develop the final product, but the prototype is thrown away.



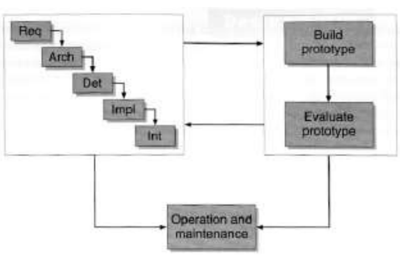
Prototyping

- Incremental
  - The final product is built as separate components, one at a time. There is one overall design for the final product, it is partitioned into independent and smaller components. The final product is then released as a series of products, each subsequent release including one more component.



Prototyping

- Evolutionary
  - The prototype is not discarded and serves as the basis for the next design iteration. The system is seen as evolving from a very limited initial version to its final release.



Papel prototyping

- Interactive paper mock-up
  - Sketches of screens appearance
  - Paper pieces showing interface elements, such as windows, menus, dialog boxes
  - Allows for the expression of the first design ideas.
- Natural interaction
  - Point with a finger → mouse click
  - Write → typing
- The designer simulates the computer's behaviour
  - Rearranging the interface elements
  - Writing answers
  - Describing effects that are difficult to demonstrate on paper
- Low fidelity in look and feel
- High fidelity in the number of features and implementation degree (depth – person simulates the computer).

Papel prototyping

- Why?
  - Faster to build (Sketch vs. program)
  - Easy to change
    - During user tests or between user tests
    - No code investment, can be thrown away
  - Focuses on the global design
    - Designer don't waste time on details
    - User makes more creative comments and suggestions (less reluctant in asking for changes)
  - Allow parallel design
  - Everyone can make a contribution

Papel prototyping

- Tools
  - Poster paper
  - Paper (A4)
  - Post-it
  - White correction tape
  - Overhead transparencies
  - Pens, pencils, scissors, tape
  - Photocopier

Papel prototyping

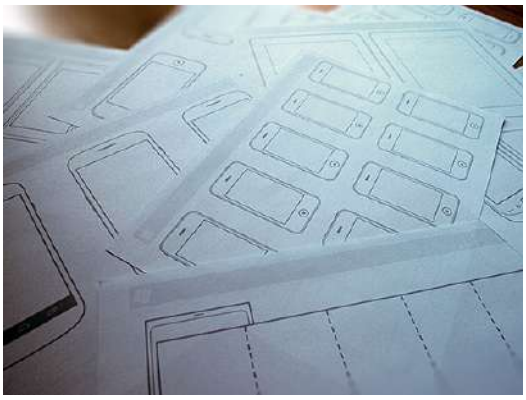
- Building
  - Bigger than real size
  - Markers are better than pencil
  - Monochrome
  - Large fonts sizes and dark lettering
  - Replace visual effects by audible explanations
    - Tooltips, animations, progress bar: 20%, 50%, 100%).
  - Keep pieces organized
    - Folders and envelops
  - ... Rehearsal

Papel prototyping

- Building

  – Hand-sketching or computer drawn (or an hybrid approach)?

Papel prototyping



•https://www.mockplus.com/blog/post/iphone-wireframe-template

Papel prototyping

- Test
  - Roles for design team:
    - Computer
      - Simulates the prototype
      - Doesn't give any feedback that the computer wouldn't
    - Facilitator
      - Presents the interface and the tasks to the user
      - Encourages the user to think aloud by asking questions
      - Leads the test
    - Observer
      - Makes no comments
      - Observes and takes notes

Papel prototyping

- Organizing the results
  - Sort comments and suggestions by priorities
  - Create a written report with the results
  - Move to the next cycle: Change the design accordingly …

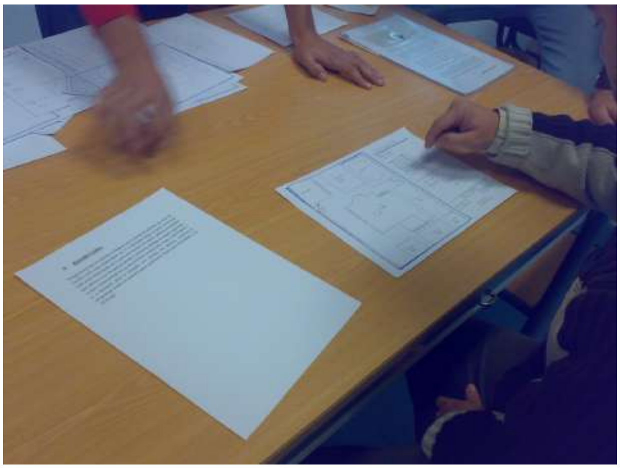
Papel prototyping

- Results
  - Functionality
    - Does it do what is needed? Missing features?
  - Navigation and task flow
    - Can users find their way around?
    - Are information pre-conditions met?
  - Terminology
    - Do users understand labels?
  - Screen contents
    - What?
    - Where?
    - Grouping

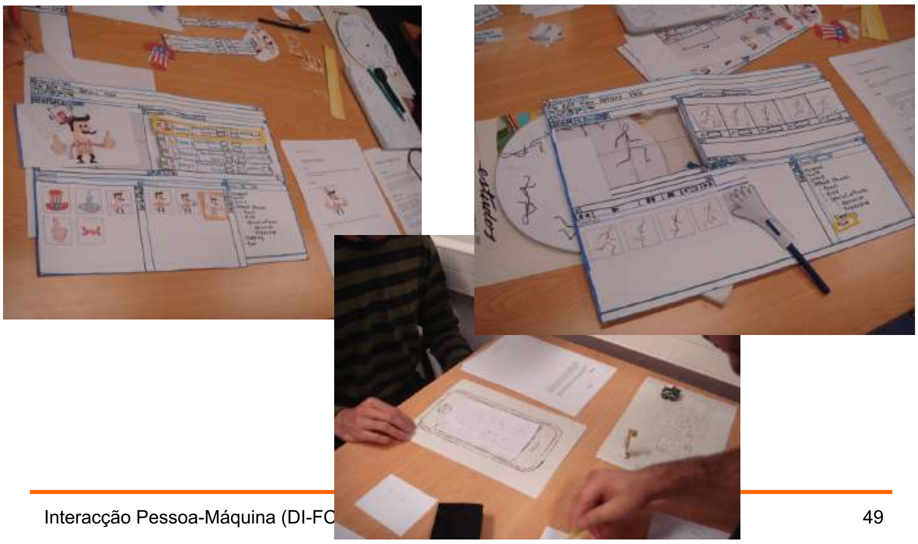
Papel prototyping

- Limitations
  - Look: colour, font, ...
  - Feel: Fitt's law issues
  - Response time
  - Dynamic feedback: animations, progress bar, mouse-over events,...
  - Context of use
  - Subtle feedback
    - Even the more subtle change in a paper prototype is noticed by the users.
  - Users tend to think much more before acting when interacting with a paper prototype.

Papel prototyping



Papel prototyping



Papel prototyping

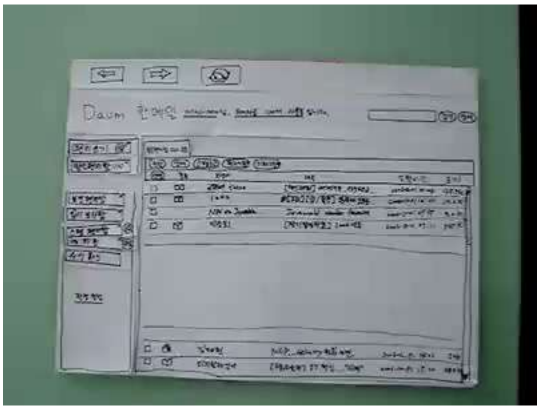
Hanmail paper prototype
http://www.youtube.com/watch?v=GrV2SZuRPv0

Trouble paper prototype
http://www.youtube.com/watch?v=dTR7gbsF7Os

Paper protype usability test
http://www.youtube.com/watch?v=ppnRQD06ggY

UX Design TimeOut and Primark
https://www.youtube.com/watch?v=3oBebzxbnGk

Papel prototyping



Hanmail paper prototype
http://www.youtube.com/watch?v=GrV2SZuRPv0

Papel prototyping

[Trouble paper prototype](http://www.youtube.com/watch?v=dTR7gbsF7Os)

http://www.youtube.com/watch?v=dTR7gbsF7Os

0

Papel prototyping

[Paper protype usability test](http://www.youtube.com/watch?v=ppnRQD06ggY)

http://www.youtube.com/watch?v=ppnRQD06ggY

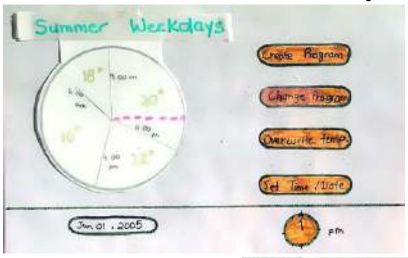
Papel prototyping

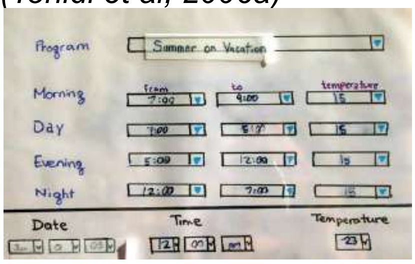
[UX Design TimeOut and Primark](https://www.youtube.com/watch?v=3oBebzxbnGk)

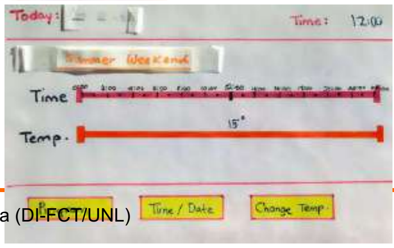
https://www.youtube.com/watch?v=3oBebzxbnGk

Paper interfaces

- *Home climate control system (Tohidi et al, 2006a)*







From Tohidi et al, 2006a

Paper interfaces

- *Distinct design language…*
- *… but same level of resolution and same functionality*

- *A group of users performed the same set of tasks on all interfaces (distinct order)*
- *Other groups saw only one interface*
- *They all performed the same set of tasks and answer the same questionnaire*

Paper interfaces

- *Paper interfaces allows parallel testing of alternatives. they make it affordable to make and compare alternative design solutions through the design process.*
- ***We should not commit to a design too soon***
- *Comparing the ratings given to the lowest rated interface of the three, as judged by users who had seen all three, that rating was significantly lower than the rating given to that same interface by users who saw only that interface.*

Paper interfaces

- *People are reluctant to be critical of designs*
- *It is easier to rate several designs (comparing) than to rate one single design solution.*
- *Constructive solutions… (Tohidi et al., 2006b)*
  - *ask users to make a simple sketch of their ideal home climate interface*
  - *users have original ideas about alternative designs*
  - *…let them communicate them in the appropriate language.*

Paper interfaces

- *Quickly explore a concept and show it to colleagues*
  - *designer is both the user and the facilitator*
  - *changes on-the-fly, based on comments*

Sketching

- *Informal testing*
  - *designer as facilitator and a representative user*
  - *changes on-the-fly, based on comments*

Sketching

- *Usability testing*
  - *more to uncover errors and determine usability than to come up with new design concepts*
  - *several users*
  - *interface can't be changed during test*

Prototyping

Paper interfaces

- *Sketches are not prototypes (remember sketches attributes)*

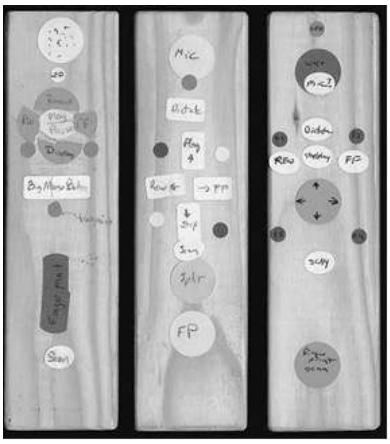
- *Sketching ≠ using inexpensive prototypes to do usability engineering.*

- *Sketching is what you use, how, when, where and why you use it.*

Dynamic paper interfaces

Physical mock-ups

3 versions of a remote control



Mixing styles

The limiting factor is your imagination…

There is always a way to express an idea appropriately within your means.

Phone graffiti

Hybrid photo-graphic composition

The main drawbacks of conventional sketching has to do with its limitations in ***capturing time and dynamics****– temporal experience related issues.*

Interaction dynamics

Make three sketches that capture:

1. *the physical nature of your mobile phone*

Easy

2. *the behaviour of the user interface of your mobile phone*

Difficult

3. *the experience of using your mobile phone*

Almost impossible

Interaction dynamics

- *We need to use more than one image to tell the story*
  - *Storyboard and comics (image sequence)*
  - *State transition diagram (global view)*
  - *"PowerPoint" slide show*

Interaction dynamics

- *Storyboard – time is distributed in space and you can see all screens simultaneously.*

- *Slide show – screens are seen sequentially*

Interaction dynamics

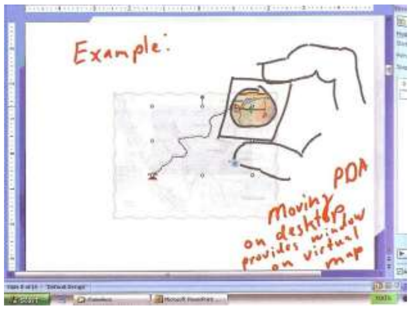
<u>Transitions</u>

- *use of arrows to describe the motion*

- *Why not just use video or animation?*
  - *dynamic is much better captured, but….*
  - *comics/storyboard approaches are faster, cheaper and enable to explore more alternatives in a given amount of time.*

Interaction dynamics

- *picture-driven-animation (Baecker, 1969)*
  - *use hand-draw line to define both objects and motion paths along which those objects move.*
  - *using this technique in PowerPoint to animate a sketch of Fitzmaurice Chameleon technique to view a map.*



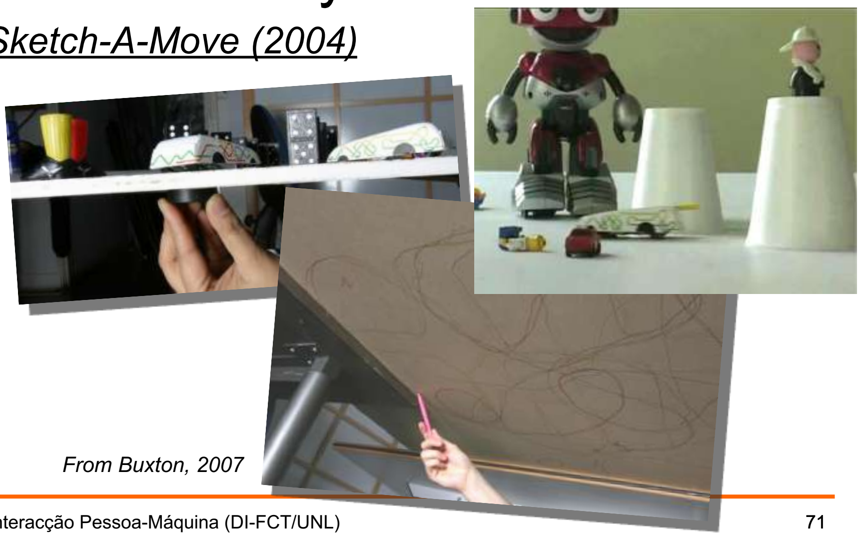
Prototyping Interaction Dynamics

Tag Around

(Duarte Gonçalves, Margarida Piriquito, Nuno Valente)

Prototyping Interaction Dynamics

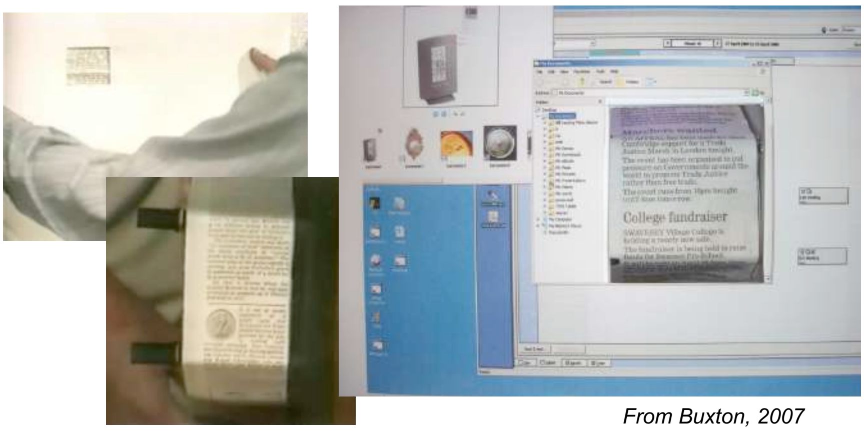
Sketch-A-Move (2004)



From Buxton, 2007

Prototyping Interaction Dynamics

Bifocal Display *(Apperley and Spence, 1982)*

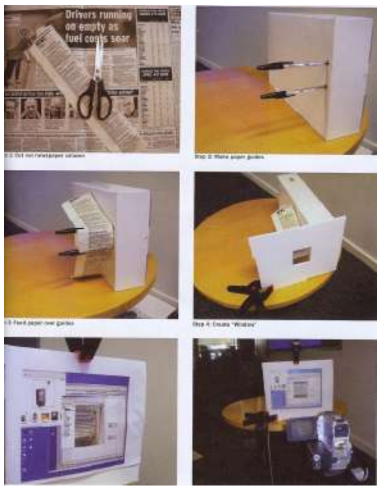


From Buxton, 2007

Prototyping Interaction Dynamics

Bifocal Display
(Apperley and Spence, 1982)

http://www.youtube.com/watch?v=qlzlA17ZN5o



From Buxton, 2007

Interaction dynamics

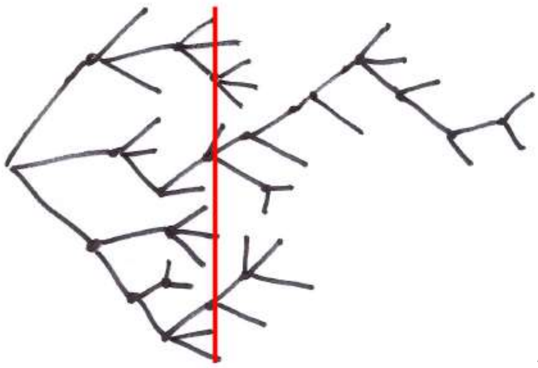
Bifocal Display *(Apperley and Spence, 1982)*

This way you can explore different aspects of the concept by manipulating:

- *% flat display vs % receding display*
- *the impact on different media (text, images,…)*
- *Horizontal scroll vs vertical scroll*
- *Receding angle*

Design is choice

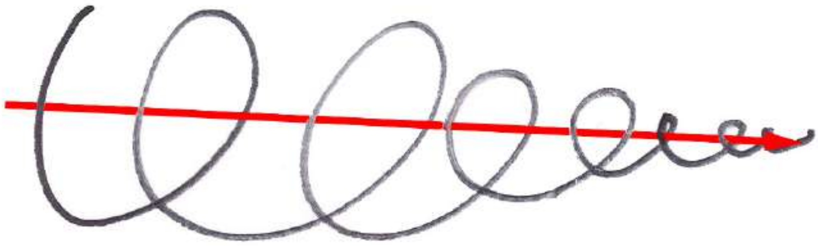
- *The purpose of design is to establish the trajectory*
- *Precedes usability engineering and is complement to it*



From Buxton, 2007

Usability engineering

- *The trajectory is already established (by the basic design already done)*
- *Iterative process that converges on a complete product (iterations allow testing and refinement of implementation)*



From Buxton, 2007

Design / Usability

- *The role of **design** is to **get the right design***
- *The role of **usability engineering** is to **get the design right***

Computational prototypes

- Interactive simulation of software
- High-fidelity look and feel
- Horizontal (low-fidelity in depth)

Computational prototypes

- Results
  - The same as from the paper prototypes + :
    - Efficiency
      - Control size, location,…
    - Screen layout
      - Is it clear, overwhelming, complicated?
      - Interface components are easy to find and distinguish?
    - Colours, fonts, icons, ...
      - Appropriate?
    - Interactive feedback
      - Do users notice the status bar messages, cursor changes or other interface feedback?

Computational Prototyping

- Prototyping techniques
  - Storyboards tools
    - Sequence of screen, possibly connected by hyperlinks
  - Form builders
    - tool for drawing real, working interfaces by dragging widgets from a palette and positioning them on a window.
  - Wizard of OZ
    - Human operator simulates and controls the system in the backstage.

Computational Prototyping

- Storyboards tools
  - **Marvel** - https://marvelapp.com/
  - Figma - https://www.figma.com/
  - Justinmind - http://www.justinmind.com/
  - Mockingbird - http://gomockingbird.com/mockingbird
  - Balsamiq - http://balsamiq.com/products/mockups

Computational Prototyping

- Prototyping techniques
  - Storyboards
    - Advantages
      - You can create anything, be creative
    - Disadvantages
      - No text entry
      - Widgets aren't functional
      - "hunt for the hotspot"

Computational Prototyping

- Prototyping techniques
  - Form builders
    - tool for drawing real, working interfaces by dragging widgets from a palette and positioning them on a window.
    - HTML
    - Java GUI builders
    - Visual Basic
    - Mac Interface Builder
    - Qt Designer
    - …

Computational Prototyping

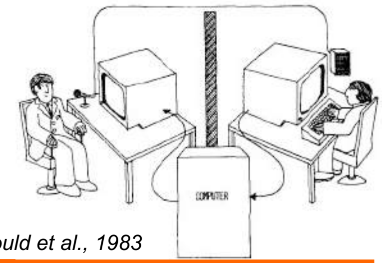
- Prototyping techniques
  - Form builders
    - Advantages
      - Working widgets, not just static pictures
      - Implementation languages allow backend development
    - Disadvantages
      - Limits creativity (fixed palette of standard widgets)
      - Not so useful for prototyping rich graphical interfaces.

Wizard of OZ

- Human operator simulates and controls the system in the backstage.
- "Wizard of OZ" = "man behind the curtain"
  - The wizard is hidden
- Create systems that let users have a real and valid experience before the system exists.
- The person using it is unaware that the system functions are performed by a human operator hidden "behind the curtains".

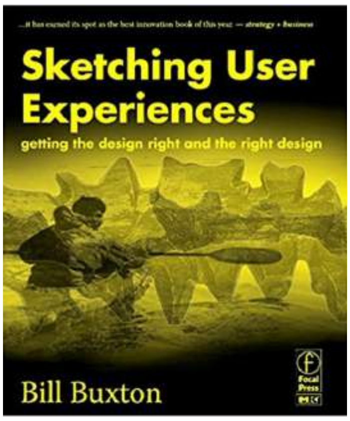
Wizard of OZ

- Often used to test new or future Technologies
  - Gould, Conti and Hovanyecz, Composing letters with a simulated listening typewriter, CACM v.26, n. 4, Abril, 1983.
  - Suede - A Wizard of Oz Prototyping Tool for Speech User Interfaces
  - OMS
- 2 interfaces (user and wizard)



From Gould et al., 1983

Further reading



References

- Apperley, M. D. and Spence, R., A bifocal display technique for data presentation. Proceedings of Eurographics' 82, 1982, pp. 27-43.
- Baecker, R., Picture-driven-animation. Proceedings of the AFIPS 1969, pp. 273-288.
- Bayles, D. and Orland T., Art Fear: Observations On the Perils (and Rewards) of Artmaking, Image Continuum Press, 2001.
- Buxton, Bill, Sketching User Experiences, Morgan Kaufmann, 2007.
- *Dix, Alan, Finlay, Janet, Abowd, Gregory, Beale, Russel. Human-Computer Interaction. Prentice Hall Europe, London, 1998.*
- *Gonçalves, D., Jesus, R., Grangeiro, F., Romão, T. and Correia N., Tag Around: A 3D Gesture Game for Image Annotation, in Proceedings of ACM ACE 2008, Tokyo, Japan, 3-5- December, 2008, pp. 259-262.*
- Gould, Conti and Hovanyecz, Composing letters with a simulated listening typewriter, CACM v.26, n. 4, Abril, 1983.
- *Pugh, S. Total design: Integrated methods for succefull product engineering, Reading MA: Addison-Wesley,1990.*

References

- Klemmer, S. R., Sinha, A. K., Chen, J., Landay, J. A., Aboobaker, N., and Wang, Suede - A Wizard of Oz Prototyping Tool for Speech User Interfaces, Proceedings of UIST '00 (November 6–8, San Diego, CA), 2000, pp. 1–10.
- Gould, J., Boies, S., Levy, S., Richards, J. and Schoonard, J., The 1984 Olympic Message System: a test of behavioral principles of system design, *Communications of ACM*, v.30 n.9, 1987.
- Gould, Conti and Hovanyecz, Composing letters with a simulated listening typewriter, CACM v.26, n. 4, Abril, 1983.
- Jakob Nielsen, Usability Engineering, Academic Press,1993.
- Suwa, M. and Tversky, B. External representations contribute to the dynamic construction of ideas. In M. Hegarty, B. Meyer and N. H. Narayanan (eds) Diagrams 2002, NY, Springer-Verlag, p. 341-343.
- Tohidi, M., Buxton, W., Baecker, R. and Sellen, A., Getting the right design and the design right: Testing many is better than one. Proceedings of ACM CHI, 2006, pp. 1243-1252.
- Tohidi, M., Buxton, w., Baecker, R. and Sellen, A., User sketches: a quick, inexpensive, and effective way to elicit more reflective user feedback. Proceedings of NordiCHI, 2006, pp. 105-114.

Interação Pessoa-Máquina

Teresa Romão
tir_at_fct.unl.pt
DI/FCT/UNL

2022/2023

1

Evaluation - Dates

Tests:

- T1: November 7, 19h
- T2: December 13, 19h

Mandatory lab classes:

- Prototype testing day: October 18, 19 and 20
- Heuristic evaluation
- Project presentation

2

Testing Day

October 18/19/20

Desire paths: the illicit trails that defy the urban planners

https://www.theguardian.com/cities/2018/oct/05/desire-paths-the-illicit-trails-that-defy-the-urban-planners

6

The Psychology of Everyday Things

Norman, D., The Design of Everyday Things. MIT Press, 1998.

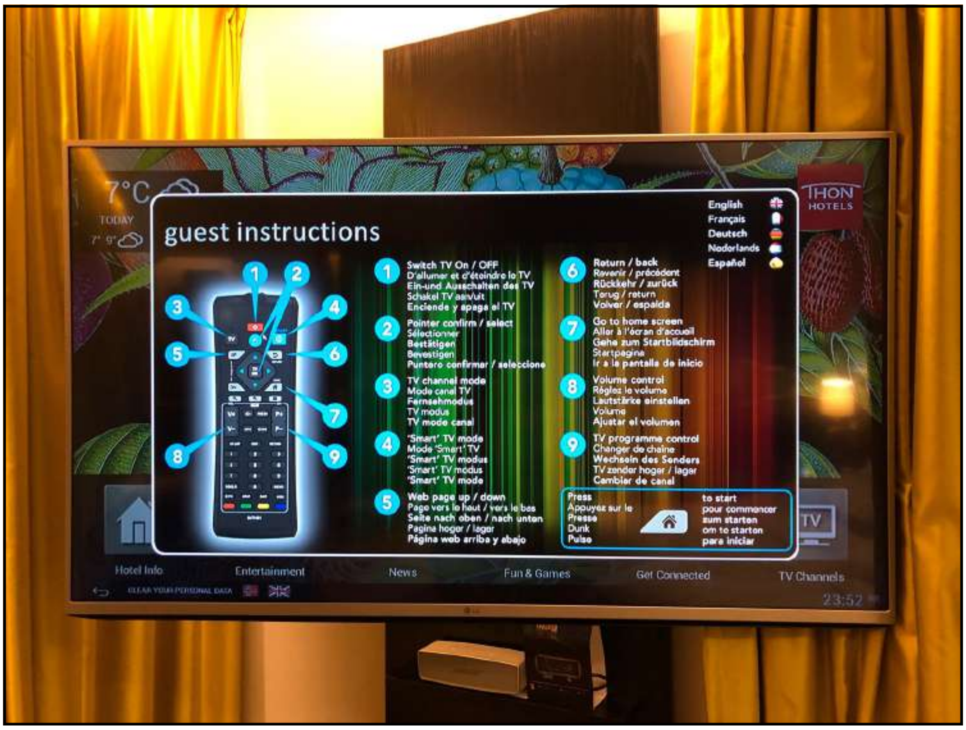
7

The Pathology of Everyday Things

- First agricultural tractors:
  - high center of gravity and short width between axles
  - rough terrain → accident!
  - Human error?
  - Probably, design error → modern tractors have low center of gravity and large rear axle.

- Everyday frustrations
  - Are you able (without reading the users manual) to use every function of your:
    - digital clock
    - mobile phone
    - Microwave
    - TVBox...

8



9

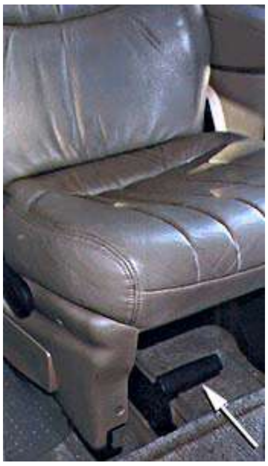
The Pathology of Everyday Things

- "Leitz Pravodit" slide projector
  - Only one button to control the presentation
  - During presentation, the slides sometimes go forward and sometimes backwards!
  - If we have access to the user manual:
    - brief push of the button → slides go forward
    - long push of the button → slides reverse
  - What an elegant design!?!
    - 1 button => 2 opposite functions
  - How was a first time user of the projector supposed to know this?!??

The Pathology of Everyday Things

- Amphitheatre Louis-Laird in Sorbonne
  - Electric projection screen: must be controlled from a separate room.
  - Why can't the person trying to lower or raise the screen see what he is doing?

The Pathology of Everyday Things



- Imagine you are on the seat shown in the figure and you pull the lever pointed by the write arrow. What do you expect to happen?

  - The seat may slide back or forward according to the force you apply.

The Pathology of Everyday Things

- Surprise!, ...the seat ejects, so it can be removed.

- Most people wouldn't expect this result, which can be dangerous.

<u>Design suggestion</u>:

  – Nobody wants to eject the seat while sitting on it. So, the control should be moved to a different position, unreachable by someone sitting on the seat.

The Pathology of Everyday Things

- This tap design is so bad that it requires instructions to be appropriately used:
  - common device
  - uncommon functioning
  - hidden control
- <u>Design tip:</u>
  - often, when a common device requires instructions of use …it means there are design problems.

15

The Pathology of Everyday Things

16

The Pathology of Everyday Things

- If we put a CD in this CD player and press the “play” buttton nothing happens. ???

The Pathology of Everyday Things

- The control buttons for the CD player are next to the cassette player and vice-versa.
- Design suggestion
  - **People expect to find the controls for a certain device next to the device they control. That’s how it should be!**

The Pathology of Everyday Things

- The figure shows a control button for a desk lamp.
- The button has 3 positions: low intensity, (I), off (O) and high intensity (II).

- What is wrong?
  - To change the light intensity, one has to turn it off first.
  - It becomes difficult to compare the 2 different states (I) e (II).

19

The Psychopathology of computers

- **Intend to type: `rm *~` to remove Emacs backup files.**

- **Actually type: `rm * ~` which removes everything!**

- **And there was no undo ...**

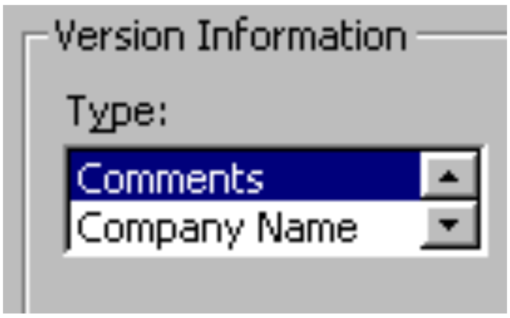
23

The Psychopathology of computers

- Reported in [Lee, 1992]:
  - In 1988, the USS *Vincennes* shot down an Iran Air A-300 Airbus with 290 people aboard.
  - The Aegis weapons system aboard the *Vincennes* had sophisticated software for identifying and tracking potential targets.
  - However, the large-screen display did not show altitude information - altitude had to be read from separate consoles.
  - The Airbus, which had levelled off at 12500 feet, was taken to be an F-14 fighter descending from 9000 feet.
  - Ironically, an escort ship with older equipment was able to read the plane's altitude quite correctly, but could not intervene in time.

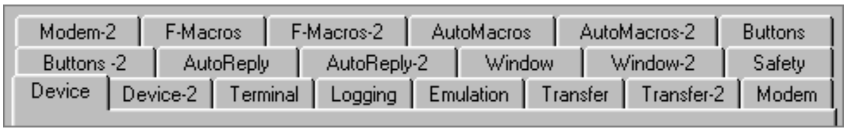
Interface Hall of Shame

- **Visual Basic 5.0 uses a list box with only 2 items (!)**
- **Radio buttons would be better.**



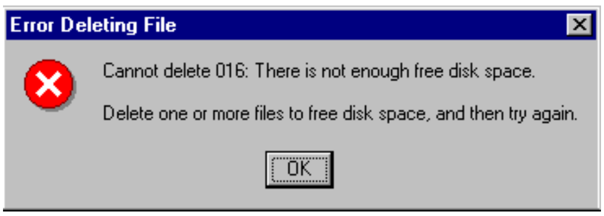
Interface Hall of Shame

- **Single-row property sheets (tab controls) are among the best interface elements.**
- **Multi-row tab controls are maybe the worst interface elements.**
- **Clicking on one of the tabs from other than the front row causes a major reorganization of the entire set of tabs.**



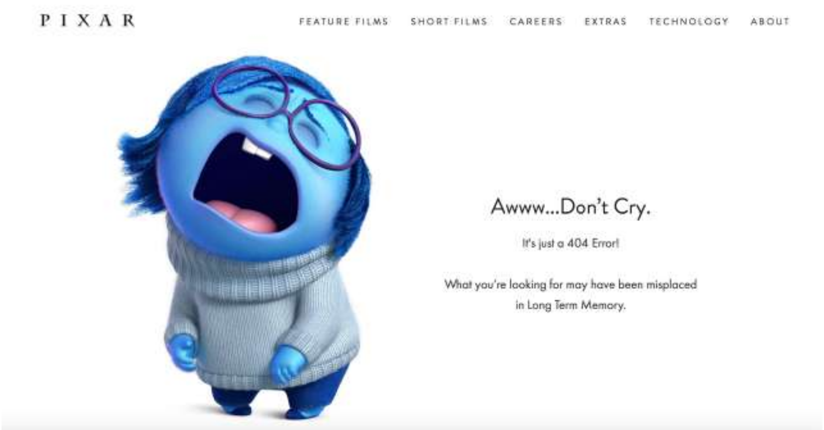
27

Interface Hall of Shame

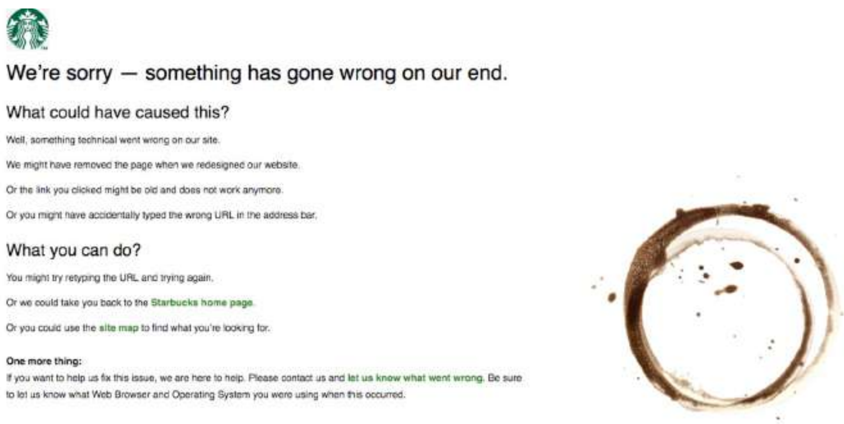


28

Error messages



Error messages



Conceptual models

- People form mental models about the way objects work, events take place or people behave – *Conceptual models*.

- Conceptual models come from:
  - Causality
  - Familiarity with similar devices
  - Experience and training
  - Instructions
  - Interaction

Conceptual models

- A **good conceptual model** allows us to
  - understand the relationship between the controls of a device and the outcome.
  - predict the effects of our actions.

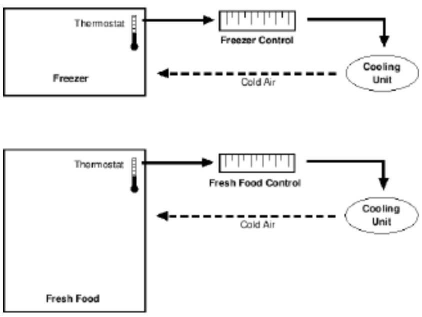
- A **poor conceptual model makes it difficult to**
  - figure out what to do in novel situations.
  - understand cause/effect.
  - predict the effects of our actions.

Conceptual models

- Example: Refrigerator
  - 2 compartments
    - freezer
    - refrigerator
  - 2 things to do:
    - adjust the temperature of the freezer compartment
    - adjust the temperature for fresh food compartment
  - 2 controls

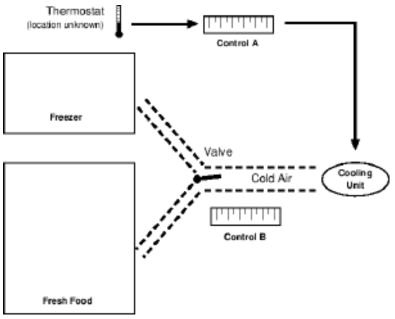
Conceptual models

- the 2 controls suggests a conceptual model to operate the two-compartment refrigerator.
- the apparent conceptual model, resulting from the controls on the device and instructions, suggests that each control is responsible for the temperature of the compartment that carries its name.



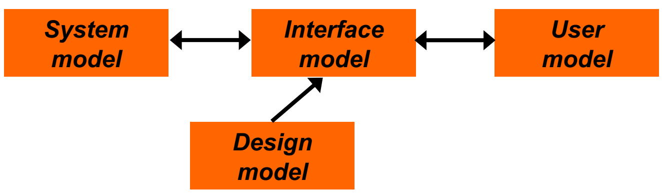
Conceptual models

- …but this conceptual model does not correspond to the way the real device works.
- the real conceptual model: there is only one thermostat and only one cooling mechanism. One control adjusts the thermostat setting, the other the relative proportion of cold air sent to each of the two compartments.



Conceptual models

- *System model* (or implementation model) = how the system works
  - its constituent parts and how they work together to do what the system does
- *Interface model* (system image) is the model that the system presents to the user.
- *User model* (mental model) is how the user thinks the system works.
- *Design model* (conceptual model) is the model that UI designer intended for the interface to convey.



Conceptual models

- The designer expects that the user model would be similar to the design model.
- But,... the design doesn't communicate directly with the user.
- Communication is done through the interface model.
- The interface model should make the design model clear and consistent to the user (avoiding that the user creates a wrong conceptual model).

41

Conceptual models

- Interface model should be:
  - Simple
  - Appropriate: reflect user's model of the task (learned from task analysis)
  - Well-communicated

42

Conceptual models

- The interface model might be quite different from the system model.

- Different system model
  - → Copper circuit ➔ cells
- Similar simple interface model

Conceptual models

- Designer's tasks:

  1. Choose the apropriate conceptual model.
  2. Correctly communicate it to the user.

How to communicate the model to the user?

- Affordances
- Mapping
- Visibility
- Feedback
- Constrains

Norman's design principles

Affordances

- Affordances - perceived and actual properties of an object that determine how the object could be used.
  - Appearance may suggest the use:
    - chair is for sitting
    - button is for pushing
    - listbox is for selection
    - knob is for turning

Affordances

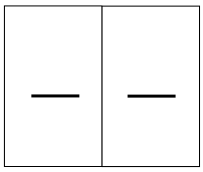
– Appearance should be used to tell the user what to do.

– The parts of a user interface should agree in perceived and actual affordances.

– When simple things need instruction, the design has failed!

Affordances

- Door
  - Can be opened and closed
  - How?
    - Pushing?
    - Pulling?
    - Sliding? Which direction?



  - The answers should given by the design, with no need for words or symbols, with no need to try.

Affordances

- Good design... the vertical bar suggests pulling; the horizontal bar suggests pushing.

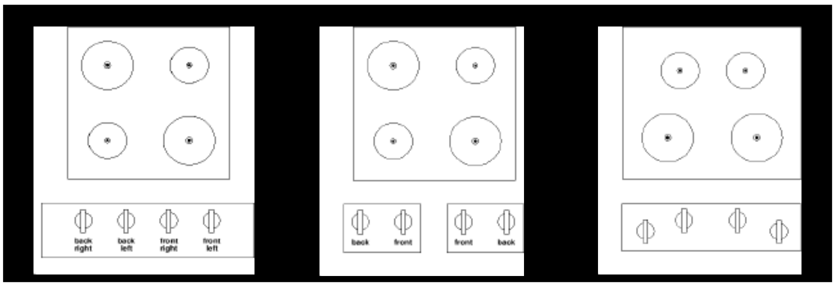
Affordances

- British Rail shelters with glass walls were vandalized routinely
  - Glass suggests ("affords") being broken
- Glass was replaced by strong plywood or concrete, and demolishing stopped
  - Wood and concrete suggests/affords stability and support
- Now, they are being scribbled
  - Smooth, even surfaces "afford" drawing

Mapping

- Mappings – relationship between the controls and their effects on the system.
- Natural Mappings – uses physical analogies and cultural conventions.
- Principle of natural mapping: the relationship between the controls and actions should be clear to the user.
- Ex:
  - to turn the car to the right, ones turns the steering wheel clockwise (its top moves to the right)
    - which control affects the direction
    - which direction to turn the steering wheel
  - Move the control up to move an object up.
  - A louder sound to represent a bigger quantity.

Mapping



- Arbitrary mapping: several possible arrangements; need for labels or memory.
- Partial mapping: only(!) 4 possible arrangements, but confusion is still possible.
- Natural mapping: no ambiguity, no need for labels, learning time or remembering.

53

Visibility

- Visibility is all about how clearly the user sees the state of the interface and all the possible actions.
- Relevant parts of the system must be visible.

54

Visibility

- Possible actions: things the user can do in the interface

- State: current configuration of the interface and its backend (ex: which objects are selected).

Hiding certain action can be advantageous. Certain functions can be kept invisible until needed
(ex: Google search)

Feedback

- Causality exists when something that happens after an action appears to be caused by that action.
- Two types of false causality:
  - Coincidences
    - Touch the keyboard just when the computer fails makes us feel guilty.
    - Run a new application just before the computer crashes.
  - Invisible effects generate confusion
    - When an action has no visible effect, we may conclude that it was ignored and we repeat it.
    - Repeatedly clicking a button with no noticeable system change.
  - Need for FEEDBACK!!!

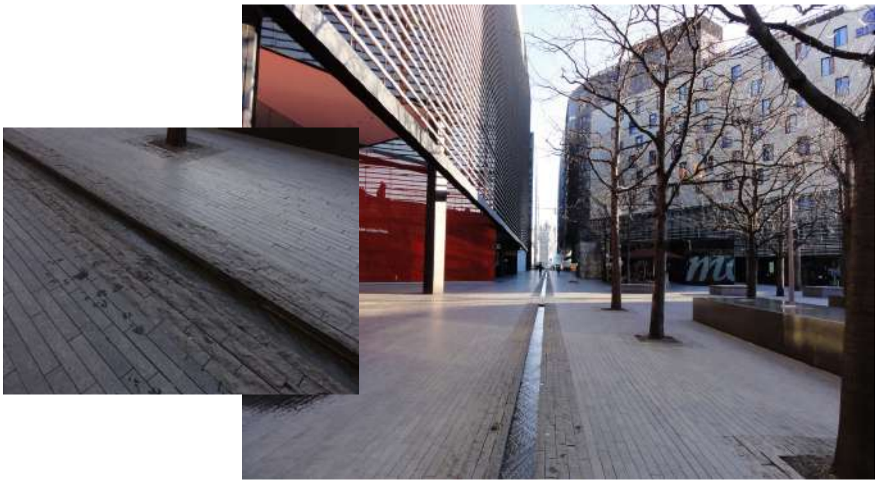
Feedback

- Actions should have an immediate visible feedback.
- Synchronized with the user action

Feedback

- Imagine:
  - trying to talk without hearing your own voice
  - trying to draw with a pencil that leaves no mark
- Types of feedback:
  - Visual
  - Audio
  - Haptic
- Examples:
  - Scrollbar thumbs move.
  - Dragged objects follow the cursor.

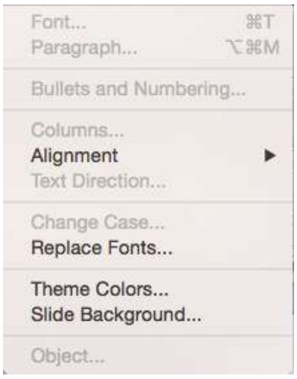
Feedback



Constraints

- + possibilities => + difficulty to manage new situations.
- Constraints restrict the number of possibilities.
- Types:
  - Physical: based on object shape; limit possible operations; more effective when visible.
  - Semantic: rely upon our knowledge of the situation and of the world.
  - Cultural: rely upon accepted cultural conventions.
  - Logical: explore logical relationships. Natural mapping provides logical constraints. Affordances suggest possibilities.
- Constraints: reduce the alternatives.

Constraints



Interação Pessoa-Máquina (DI-FCT/UNL) 63

63

Constraints

Interação Pessoa-Máquina (DI-FCT/UNL) 64

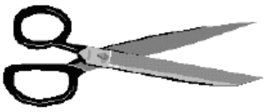
64

Constraints

- Cultural conventions may vary:
  - Light switches:
    - USA: down - off
    - UK: down - on
  - Taps:
    - USA: turn left - open
    - UK: turn left - close
  - Red:
    - USA: danger
    - Egypt: dead
    - India: life
    - China: happiness
  - ...

Conceptual models

- Scissors provide a good conceptual model:
  - Affordances
    - holes to put the fingers



  - Constraints
    - Big hole suggests several fingers and small hole for the thumb.
  - Mapping
    - between fingers and holes
- Conceptual model
  - operating parts are visible and the implications are clear. The conceptual model is made clear.

Conceptual models

- A digital clock with several buttons doesn't provide a good conceptual model.
  - Affordances
    - buttons suggest pushing,...but what do they do?
  - Mapping
    - no evident relationship between the buttons and their functions
  - No constraints
  - Former Knowledge
    - not similar with mechanical clocks.
- Conceptual model
  - Must be formed from instructions.

Metaphor

- A Metaphor
  - Another way to address the conceptual model problem
  - Relating computing to other real-world activity is an effective teaching technique
    - Desktop
    - Trashcan
  - Several good interfaces are not based on metaphors
    - Hyperlink
    - Resizeable windows

Metaphor

- Advantages
  - Help users to perceive the conceptual model
  - Help to understand the “unfamiliar”
  - Simplify the description of the system to novice users
  - Facilitate users access to computers
- You’re borrowing a conceptual model that the user already knows.

Metaphor

- Problems
  - Hard to find
  - Constraining
  - Some tasks do not fit into a given metaphor
  - Cultural differences
- Not such a good example!

- Trash bin over the desk?
- Drag to trash ⇒ Delete
- Disk eject ⇒ drag to trash !!

Metaphor

- The basic rule for metaphors is: use it if you have one, but don't stretch for one if you don't.
- Use of a metaphor is no guarantee for a good communication of the conceptual model:
  - RealCD: bad affordances, visibility

71

Metaphor

Interface Hall of Shame

72

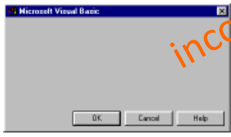
Metaphor

Consistency

- When people lack the knowledge about how to operate a certain system, they tend to derive the operation by analogy with other similar system.
- Principle of least surprise
- Similar things should look and act in similar ways.
- Different things should be visibly different

Consistency

- Internal – within itself



- External – with other application

- Metaphorical – with the interface metaphor or similar real-world objects

Knowledge distribution

- Knowledge in the head and in the world
  - partially in the head
  - partially in the world
  - partially in constraints
- Knowledge in the world
  - reduces the need for learning and mental effort.
  - Ex:
    - the interface can show the input format:
      - >Please enter the date (yyyy/mm/dd):_
      - The slots' format only allows for the correct object.

Interface Hall of Shame

Interação Pessoa-Máquina (DI-FCT/UNL) 78

Interface Hall of Shame

- **Inconsistency: the scrollbar usually serves to move the content of a window.**
- **Affordance: continuous move; not discreet selection.**
- **Frequent users have no advantages: how to find a template used before?**
- **OKAY?**
- **Text aligned to the right?**

Interação Pessoa-Máquina (DI-FCT/UNL) 80

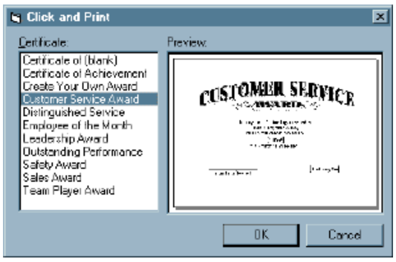
Interface Hall of Shame

- **Suggestions to correct the interface problems?**

Interação Pessoa-Máquina (DI-FCT/UNL) 81

81

Interface Hall of Shame



- **Affordance: a list box suggests the selection of an item.**
- **Random access to available templates is trivial.**
- **No need for help messages.**

Interação Pessoa-Máquina (DI-FCT/UNL) 82

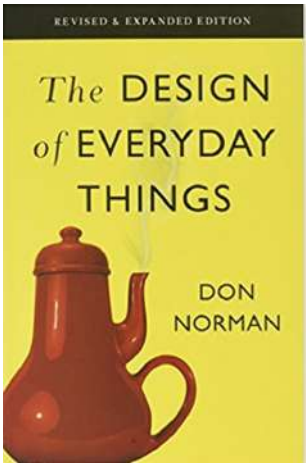
82

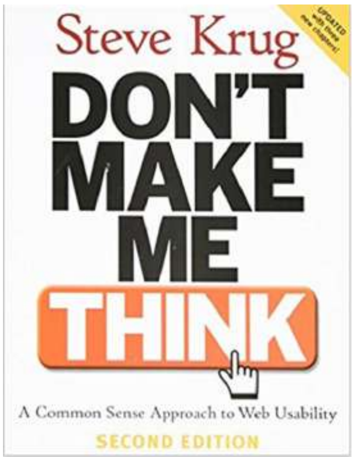
Previously correcting errors

- **A paper prototype of this interface in an initial iteration would have detected several problems. In that iteration, modifications would have cost just one more "sketch".**

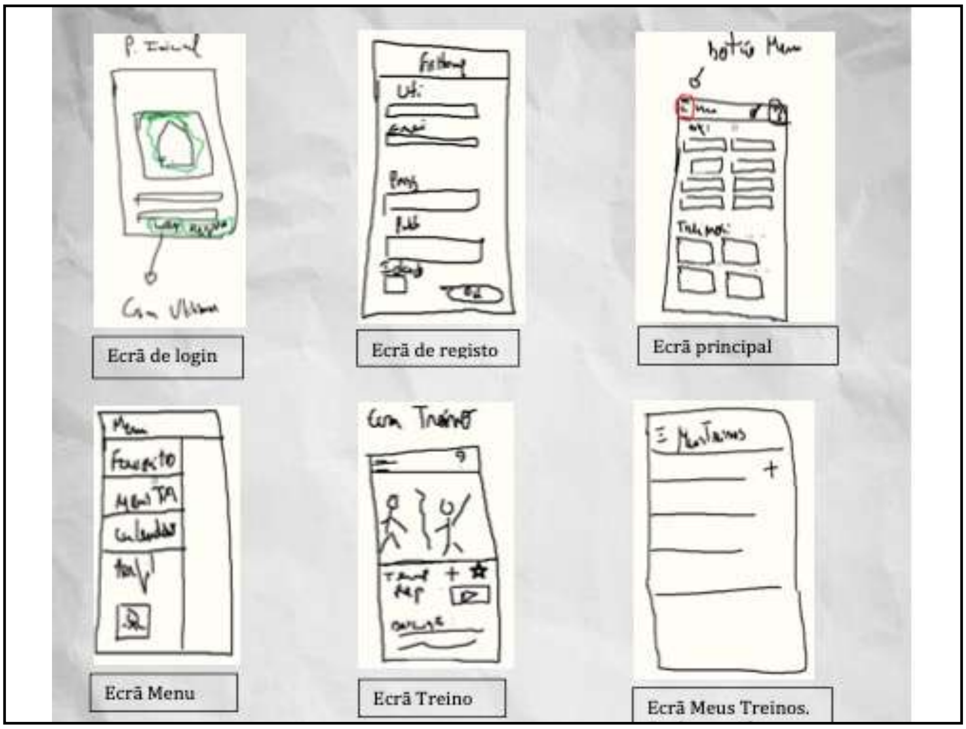
83

Further reading

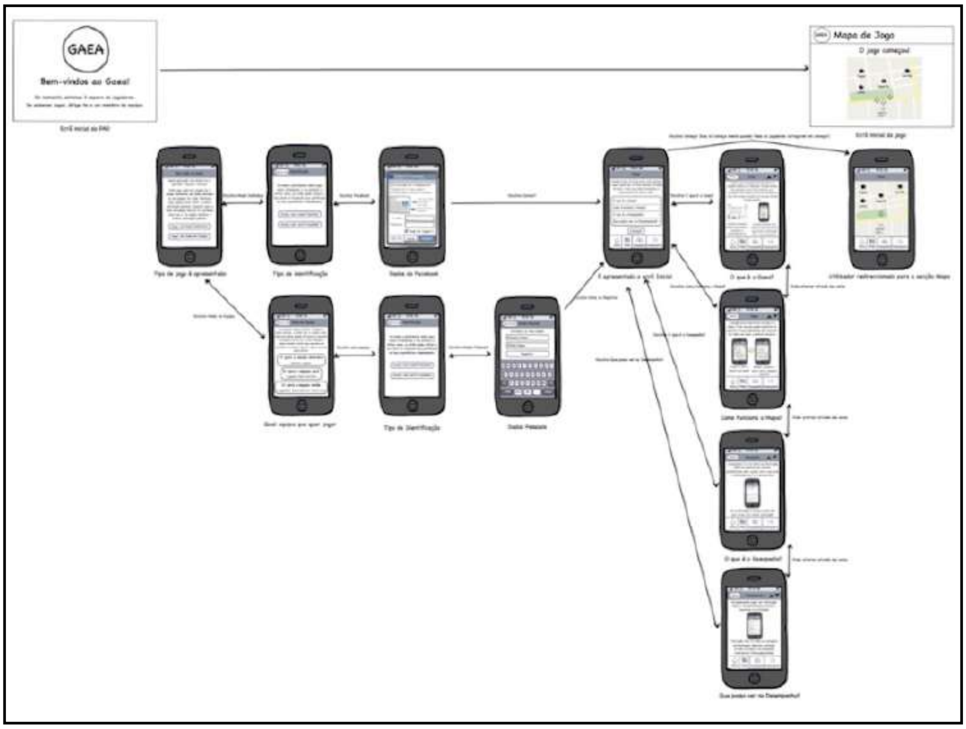




84


Ecrã de login
Ecrã de registo
Ecrã principal
Ecrã Menu
Ecrã Treino
Ecrã Meus Treinos.

85


GAEA
Bem-vindos ao Gaea!
Mapa de Jogo

86

Interaction models

Interaction - Concepts

- *What is Interaction?*
  - A process of information transfer.
  - Communication between the user and the system
    - two complex entities
    - ≠ ways of communication and view of the domain
  - The interface must effectively translate between them
    - the translation may fail for several reasons.

- *Models*
  - Interaction models help to understand what is going on in the interaction and to identify the roots of difficulties.
  - Provide a framework to compare different interaction styles.

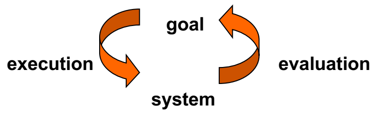
Interaction – Norman’s model

"*The basic idea is simple. To get something done, you have to start with some notion of **what is wanted**—the **goal** that is to be achieved. Then, you have to do something to the world, that is, **take action** to move yourself or manipulate someone or something. Finally, you **check** to see that your goal was made. So there are four different things to consider: the goal, what is done to the world, the world itself, and the check of the world. The action itself has two major aspects: doing something and checking. Call these **execution** and **evaluation**.*"

[Norman]

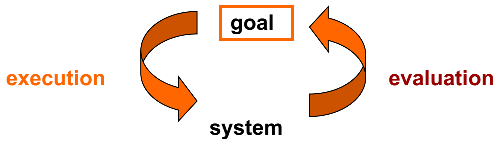
Interaction – Norman’s model

- Donald Norman’s model of interaction is the most influential in HCI, possibly because of it's closeness to our intuitive understanding of the interaction between human users and computers. It comprises 7 stages:
  - establishing the goal
  - forming intention
  - specifying the action sequence
  - executing the action
  - perceiving the system state
  - interpreting the system state
  - evaluating the system state with respect to the goals and intentions.



- Norman’s model concentrates on the user’s view of the interface.

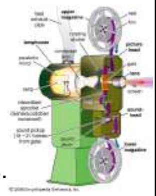
Interaction – Norman's model



- Seven stages of action:
  - establishing the goal
  - forming intention
  - specifying the action sequence
  - executing the action
  - perceiving the system state
  - interpreting the system state
  - evaluating the system state with respect to the goals and intentions.

Interaction – Norman's model

- Norman uses his model of interaction to show why some interfaces causes problems to their users.



- Some systems are harder to use than others.
- **Gulf of execution**
  - the difference between the intentions and the allowable actions.
  - user's formulation of actions ≠ actions allowed by the system
- **Gulf of evaluation**
  - reflects the amount of effort the user must exert to interpret the physical state of the system and to determine how well the expectations and intentions have been met.
  - user's expectation of changed system state ≠ actual presentation of this state

Interaction – Norman's model

- The seven stages of action prompt the following design questions:

  How easily can one:
  1. determine the function of the device?
  2. tell what actions are possible?
  3. determine mapping from intention to physical movement?
  4. perform the action?
  5. tell what state the system is in?
  6. determine mapping from system state to interpretation?
  7. tell if system is in the desired state?

Interaction – Norman's model

- **Principles of Good Design**
  The significance of these questions can be summed up as the following principles of good design:
  – *Visibility:* By looking, the user can tell the state of the device and the alternatives for action.
  – Affordances: Objects' appearance determine how the object could be used.
  – *Feedback:* The user receives full and continuous feedback about the results of actions.
  – *Good mappings:* It is possible to determine the relationship between actions and results, between the controls and their effects, and between the system state and what is visible.
  – *A good conceptual model:* The designer provides a good conceptual model for the user, with consistency in the presentation of operations and results and a coherent, consistent system image.

Norman, 1998

Dificulties in interaction

- Gulf of execution
  - Difficulty of choosing actions and performing them
  - Affordances, constraints, mappings are helpful

- Gulf of evaluation
  - Difficulty of determining the effects of your actions
  - Feedback is essential here

To err is human

Norman's thoughts about Errors

- If an error is possible, someone will make it.
  Assume: All possible errors will be made!
- **Design for error**
- Design exploratory systems, with the possibility to undo actions.

To err is human

Norman's thoughts about Errors

– Designers should:
  - assume all possible errors will occur
  - minimize the chance of errors
  - minimize the effects of errors when they do occur
  - make it easy for users to detect errors
  - make it possible to reverse the effects of an error

100

To err is human

- Mistake
  – form the wrong intention
  – often caused by a wrong conceptual model

- Slip
  – error in carrying out the intention (wrong execution)
    - Description errors
    - Capture errors
    - Mode errors

104

To err is human

- Description error
  - Intended action is replaced by another action with much in common
    - Pouring orange juice into your cereal
    - Throwing shirt into toilet instead of laundry basket
      - "throwing the shirt at the top of the container"
  - the internal description of the intention was not sufficiently precise.
  - Avoid actions with very similar descriptions
    - Long rows of identical switches

To err is human

- Capture error
  - A sequence of actions is replaced by another sequence that starts the same way. Usually the first is unfamiliar and the second is well practiced.
    - Leave your house and find yourself walking to school instead of where you meant to go
    - Vi :wq! Command

:Wq! – save & quit
:w – save
q! – quit without save

  - Avoid usual action sequences with common prefixes.

To err is human

- Mode error
  - Occurs when devices have different modes of operation and the same action has different meanings depending on the selected mode
    - Vi's insert mode vs. command mode
    - Caps Lock
  - Avoiding mode errors
    - Eliminate modes
    - Visibility of mode
    - Spring-loaded or temporary modes
    - Disjoint action sets in different modes

107

To err is human

- Lack of consistency
  - When people lack the knowledge about how to operate a certain system, they tend to derive the operation by analogy with other similar system.
  - It is a powerful method of human though,…
  - … it can lead to errors if the mapping is not consistent.

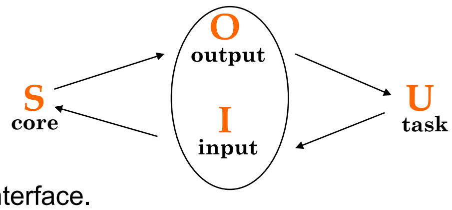
108

To err is human

- Minimize error occurrence
  - actions with serious consequences and irreversible actions should be difficult to perform and require confirmation
- Minimize the effect of errors
- Support recovering

Interaction – Interactive cycle

- Interaction Framework from Abowd and Beale is an extension of Norman's model. It attempts a more realistic description of interaction by including the system explicitly.
- Nodes represent the 4 major components of an interactive system:
  - System
  - User
  - Input
  - Output



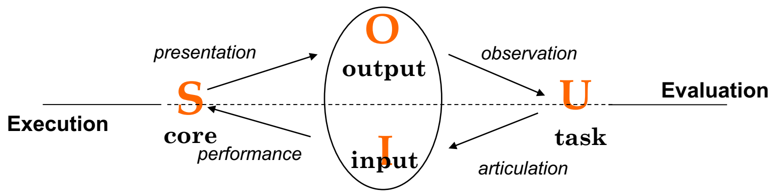
- Input and Output form the interface.
- Each component has its own language: core, task, input, output.

interaction ⇒ translation between languages

problems in interaction = problems in translation

Interaction – Interactive cycle

- 4 main steps, each corresponding to a translation from one component to another:



- 2 phases: Execution and evaluation.

Interaction – Interactive cycle

- **Execution phase**
  - User starts the interaction cycle
    - formulation of the goal
    - formulation of the task to achieve the goal
  - the only way the user can manipulate the machine is through the *Input*.
    → a task must be articulated within the *Input* language.
  - the *input* language is translated into the *Core* language as operations to be performed by the system.
  - the system transforms itself and get into a new state.
- **Evaluation phase**
  - the new system state must be communicated to the user.
  - the current values of the system attributes are translated to the Output language.
  - the user observes the Output and evaluates the results of the interaction in respect to the goal.

Interaction – Interactive cycle

- Summary of Abowd & Beale´s model:
  - user intentions
    → translated into actions at the interface
    → translated into changes in system state
    → reflected in the output display
    → interpreted by the user
- general framework for understanding interaction
  - an abstraction
  - not restricted to computer systems
  - identifies all major components involved in interaction
  - allows comparative assessment of systems

Interaction - Ergonomics

- Ergonomics - study of the physical characteristics of the interaction, which includes:
  - Control's arrangement
    - controls should be grouped logically (keeping opposing controls separate); according to function, frequency of use or sequentially.
    - the whole system interface must be appropriately arranged in relation with the user's position; the user should able to reach all controls and see all displays without excessive body movement.
    - critical information should be displayed at the eye level.
    - appropriate light should be used, not distorting the display.
    - space between controls should be adequate, in order to facilitate the user manipulation.
  - Physical surrounding environment
    - Ex: adaptable seats for all sizes of users, comfortable positions.
  - Health issues
    - physical position, temperature, lighting, noise,...
  - Use of colour
    - colour characteristics and interpretation by users; be aware of human psychological and physical characteristics, as well as cultural differences.

References

- Dix, Alan, Finlay, Janet, Abowd, Gregory, Beale, Russel. *Human-Computer Interaction*. Prentice Hall Europe, London, 1998.
- Abowd, G. D. & Beale, R. (1991) 'Users systems and interfaces: a unifying framework for interaction'. In: HCI'91: People and Computers VI. Eds: D. Diaper and N. Hammond. Cambridge University Press, Cambridge. pp. 73-87. http://books.google.com/books?hl=en&lr=&id=sex1FtiQD0oC&oi=fnd&pg=PA73&dq=%22Gregory+D.+Abowd%22+%22G.+D.+Abowd%22&ots=5iy95k0VcX&sig=EjhVOOOi5aucTnlbhow5N8p2NIc#v=onepage&q&f=false

References

- Interface Hall of Shame - http://hallofshame.gp.co.at/index.php?mode=original
- Leonard Lee. *The Day the Phones Stopped*. New York: Primus-Donald I. Fine, Inc.1992.
- **Norman, Donald. *The Design of Everyday*** *Things*. MIT Press, 1998.

Complementary reading

- Norman, Donald Human error and the design of computer systems. *CACM*, v.33 n.1, 1990.

  http://cogsci.ucsd.edu/~norman/DNMss/errordesign.html

- Norman, Donald Design Rules Based on Analyses of Human Error. *CACM*, v.26 n.4, 1983.

  http://doi.acm.org/10.1145/2163.358092

Interação Pessoa-Máquina

Teresa Romão
tir_at_fct.unl.pt
DI/FCT/UNL

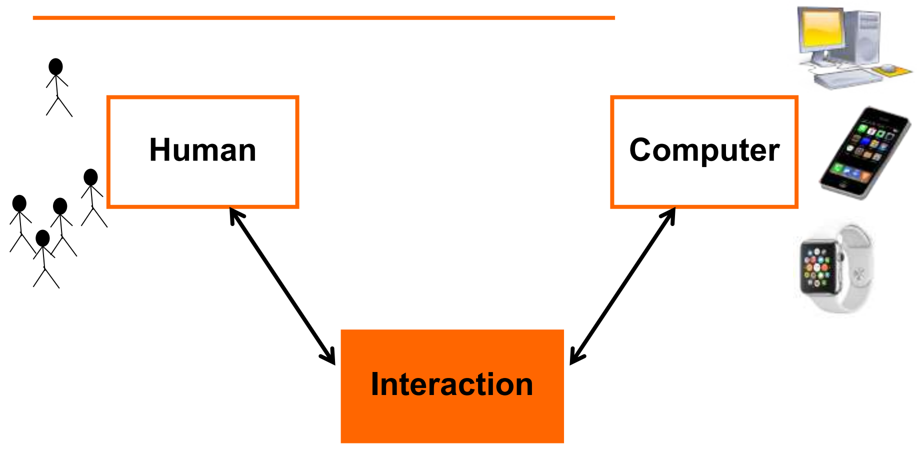
2022/2023

Evaluation - Dates

Tests:
- T1: November 7, 19h
- T2: December 13, 19h

Mandatory lab classes:
- Prototype testing day: October 18, 19 and 20
- Heuristic evaluation
- Project presentation

Interactive System components



Human

Human

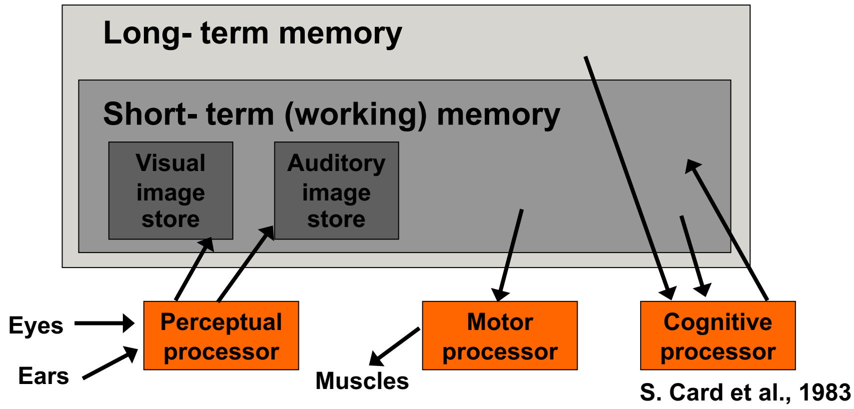
- **Human have limited capacities to process information**
- **The information is received and sent through several input/output channels:**
  - Visual
  - Auditory
  - Haptic
  - Movement
- **The information is stored in memory:**
  - Sensory memory
  - STM
  - LTM
- **The information is processed and applied:**
  - Reasoning
  - Problem solving
  - Knowledge acquisition
  - Error
- **Users share many capabilities, but, at the same time, they have many different characteristics that influence the way they interact with the surrounding environment.**

Human

- “Model Human Processor” (S. Card et al., 1983) – a simplified view of the human processing involved in interacting with computer systems:
  - Perceptual system – handle the sensory stimulus from the outside world.
  - Motor system – controls actions.
  - Cognitive system – provides the necessary processing to connect the two above.
- Processing and memory is required at all levels.
- The model includes a set principles of operation which dictate the behaviour of these systems under certain conditions.

Human

• Model Human Processor (MHP)



Human

• Model Human Processor (MHP)

– Processors' cycle time

- Tp ≅ 100ms [50-200ms]
- Tm ≅ 70ms [25-170ms]
- Tc ≅ 70ms [30-100ms]

Human

- Model Human Processor (MHP)
  - Perceptual fusion
    - 2 events (stimuli) in the same cycle time (Perceptual processor – Tp ≅ 100ms) appear fused (in the same frame).
    - Motion picture - 1/Tp frames/second are enough
    - Feedback in < Tp feels instantaneous
    - Sense of causality

Human

- A more simple model:
  - Receive information and respond through input/output channels.
  - The information is stored in memory.
  - The information is processed and applied in several ways.
- Human capabilities are relevant…
- …as well as the individual differences.

Human - I/O channels

- Input → senses
- Output → motor control

- Senses:
  - Sight, hearing, touch, smell and taste.

- Fingers, eyes, head, vocal system.

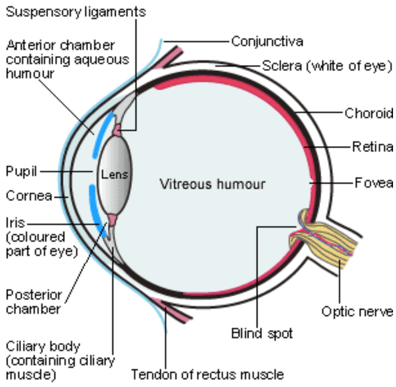
Human - Vision

- Primary source of information for the average person

- Two stages:
  - Reception of physical stimuli
  - Stimuli interpretation and processing

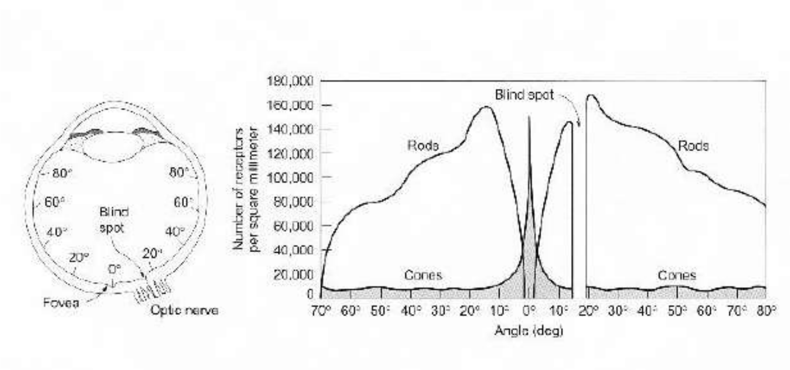
- Vision apparatus: eye
  - Mechanism that receives light and transforms it in electrical energy.
  - Light is reflected from objects; their image is focused upside down in the back of the eye.
  - The retina contains 2 types of photoreceptors: **rods**, highly sensitive to light, allowing us to see under a low level of illumination (dominate peripheral vision); and **cones**, allowing colour vision (sensitive to different wavelength of light).
  - Ganglion cells: X-cells detect patterns and Y-cells detect movement.

Human - Vision



Human - Vision

Distribution of rods and cones in the retina



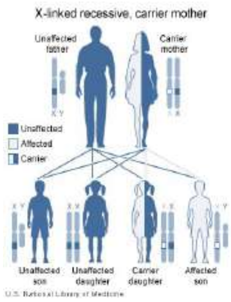
Human - Vision

- Colour
  - Cones are sensitive to light of different wavelengths. There are 3 different types of cones, each sensitive to a different colour light.
  - Only 3-4% of the fovea is occupied by cones which are sensitive to blue light (blue acuity is lower – don't use blue for small details).
  - 8% of males and 1% of females suffer from colour blindness.

Human - Vision

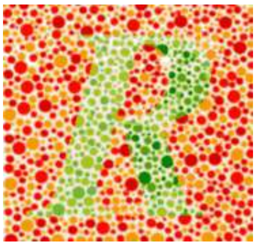
- Colour blindness
  - "Recessive" gene located in chromosome X
  - Males can only transmit colour blindness to their daughters.
  - Better night vision

Genotype	**Phenotype**
$X_D \mid X_D$	Female with normal vision
$X_D \mid X_d$	Female with normal vision
$X_d \mid X_d$	Female with colour blindness
$X_D \mid Y$	Male with normal vision
$X_d \mid Y$	Male with colour blindness

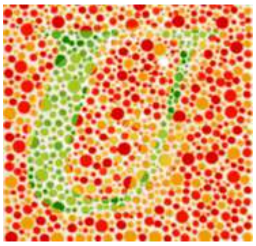


Human - Vision

- Colour Blindness
  - Ishihara test



1



2

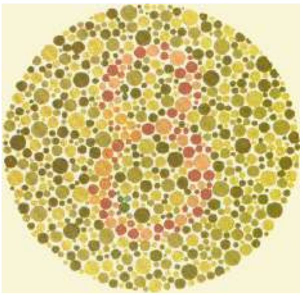
Human - Vision

- Colour blindness
  - Ishihara test

Image	Normal vision	Deficient perception of red and green	Lack of colour perception
1	R	E	--
2	U	G	--

Human - Vision

- Colour blindness
  - Ishihara test



Human - Vision

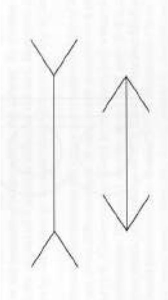
- Visual processing
  - Visual processing involves the transformation and interpretation of a complete image, from the light that is thrown onto the retina.
  - Our expectations affect the way an image is perceived:
    - If we know that an object is a particular size, we will perceive it as that size no matter how far it is from us.
  - Visual processing compensates for the movement of the images on the retina and changes in luminance.

Human - Vision

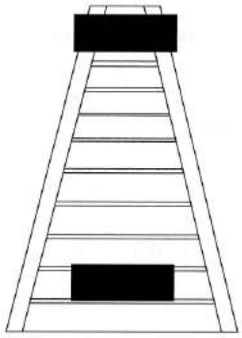
- Visual processing

  Optical illusions sometimes occur due to overcompensation

The Muller-Lyer illusion



The Ponzo illusion



Human - Vision

Human - Vision

- Perceptions
  - are not a mere sum of sensations…
  - are influenced by:
    - our current emotional state
    - the context
    - our experience
    - …

Human - Vision

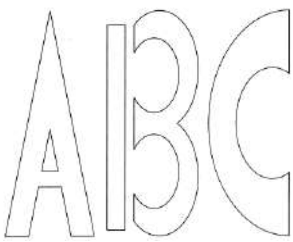
From Joel Santos, Fotografia, Centro Atlântico, 2010

Human - Vision

- Visual processing
  - Context is used to solve ambiguities

Human - Vision

- Visual processing



Human - Vision

- Reading
  - When reading we make a series of fixation-saccade-fixation sequences.
  - Reading eye movements:
    - saccades, the eye movement itself
    - fixation duration or the intersaccadic interval
    - regressions (i.e. right-to-left eye movements)
    - return sweeps (going from the end of one line to the beginning of another).
  - No information is taken in during saccades (10-25 msec), regressions (10-25 msec) or return sweeps (40 msec).
  - During fixation (250 msec) a visual pattern is reflected on the retina.

Human - Vision

- Reading
  - Several stages
    - Visual pattern perception (characters and words)
    - Decode with reference to an internal representation of language
    - Interpretation by syntactic and semantic analysis
  - Font size, spacing, line length have influence in the reading speed.
  - Adults read approximately 250 words a minute.
  - Reading from a computer screen/Book: Speed? UX?

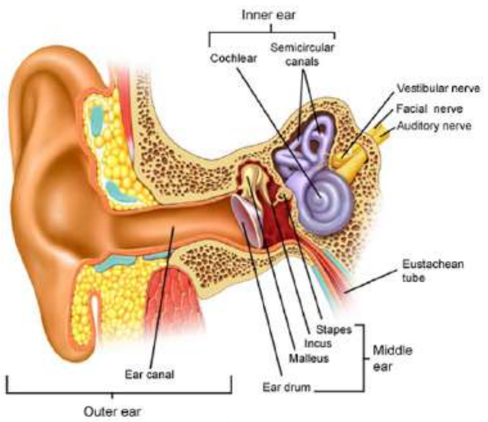
Human - Vision

- Reading
  - Font sizes of 9 to 12 points are equally legible, given proportional spacing between lines.
  - Line lengths of between 58 e 132 mm are equally legible.
  - Negative contrast (dark characters on a light screen) provides higher luminance and, therefore, increased acuity than positive contrast.

Human - Audio

- Hearing
  - Provides us with information about our environment:
    - Objects, distances, directions, ...
  - Try to close your eyes and listen:
    - What sounds can you identify?
    - Where do they come from?
  - Human ear
    - Outer ear: protects the middle ear, collects sound waves and channels them down the ear canal to the middle ear and amplifies some sounds.
    - Middle ear: transmits the sound waves, as vibrations, to the inner ear and amplifies sound.
    - Inner ear: Chemical transmitters are released and causes impulses in the auditory nerve.

Human - Audio



Human - Audio

- Sound (vibrations in the air):
  - Pitch – sound frequency
  - Loudness - amplitude
  - Timbre - type or quality
- Humans are able to identify sound's location
- Audible frequencies: 20Hz a 20kHz
- The auditory system filters the sound – we are able to distinguish sounds despite of the background noise
  - Cocktail party effect

Human - Audio

- How can we use the properties of sound, effectively, in interface design?

Human - I/O channels

- Touch
  - Provides important feedback information about the surrounding environment.
    - Catch a glass of water without feelling it.
    - Manipulation of objects in virtual reality systems.
  - It is an essential sense for visual impaired people.
  - Stimuli are received by sensory receptors in the skin.
  - Some areas of the body are more sensitive than others.
    - Two-point threshold test
  - We are aware of the position of our body and limbs (affect performance).

Human - I/O channels

- Movement
  - Movement time:
    - Stimuli reception → processing → response generation
    - Depends on physical characteristics: age, fitness, ...
  - Reaction time
    - Depends on the type of stimuli
      - visual: 200ms
      - auditory: 150ms
      - pain: 700ms
    - Combined stimuli reduces reaction time.
    - Decreases with skills and practice and increases with fatigue.

Human - I/O channels

- Movement
  - Accuracy:
    - Speed of reaction results in reduced accuracy?
      - Depends on the task and the user
      - Video gamers / Keyboard operators
  - Speed and accuracy to move to particular target on the screen (button, icon, menu item).
    - Depends on the size of the target and the distance that have to be moved.

Human - Movement

- **Fitts' Law**
  – Describes the time a user takes to select a target on the screen.
  – Time (Mt) to move your hand to a target of size S at distance D is:

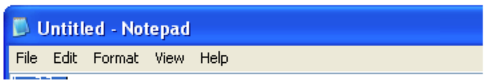
$$Mt = a + b \log_2 (D / S + 1)$$

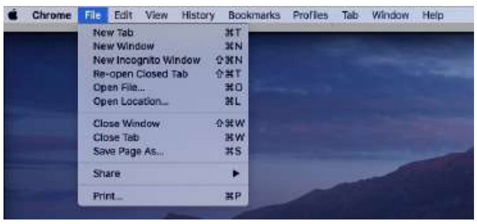
  – Mt – movement time
  – a e b – empirically determined constants
  – D - distance
  – S - size

  – In general:
    - Targets should be as large as possible
    - Distances should be as small as possible

Human - Movement

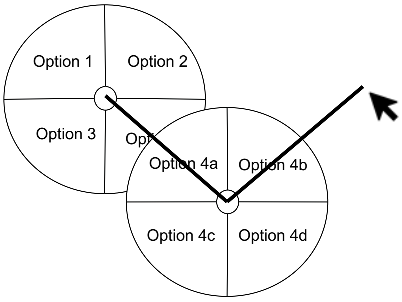
- Fitts' Law
  – Targets at screen edge are easy to hit
    - Mac/Windows menubar

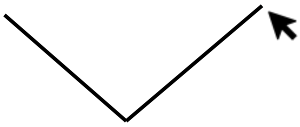




Human - Movement

- Fitts' Law
  - Linear pop-up menus vs pie menus



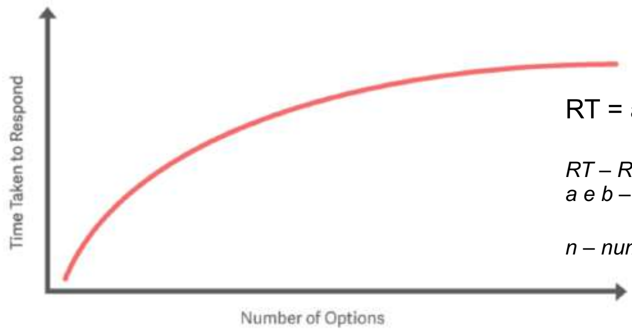


Human - Movement

- Fitts' Law
  - Hierarchical menus
    - Windows – 500ms timeout (sense of causality is lost).
    - Mac - triangular zone, spreading from the mouse to the submenu, in which the mouse pointer can move without losing the submenu.
  - Fitts's law demo (http://fww.few.vu.nl/hci/interactive/fitts/)

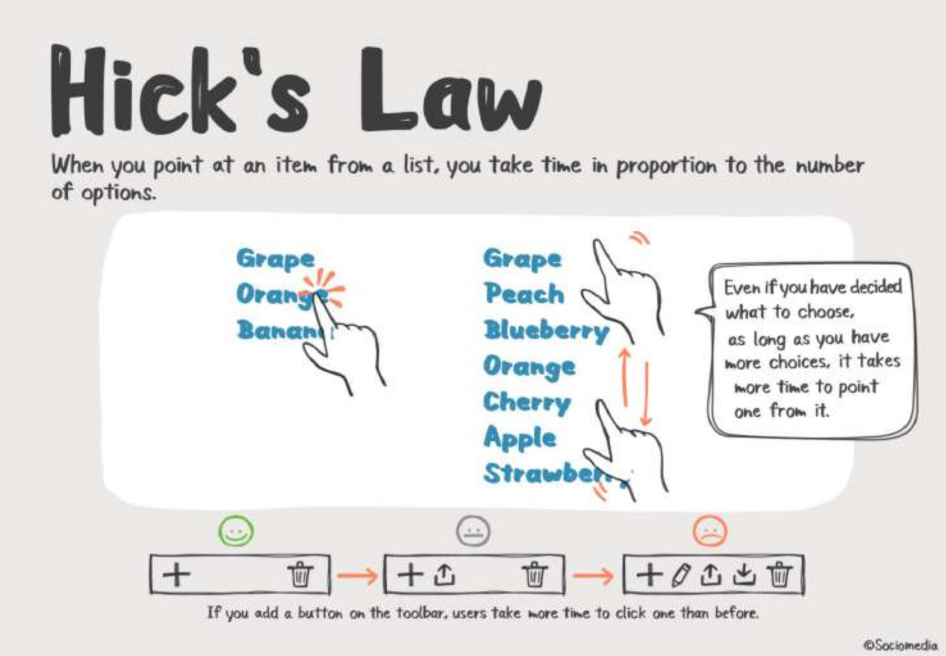
Human - Movement

- Hick's Law



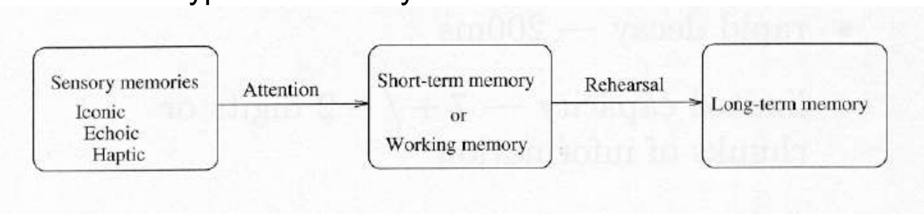
$$RT = a + b \log_2 (n + 1)$$

RT – Reaction time
a e b – empirically determined constants
n – number of options



Human - Memory

- Three types of memory:



- Sensory Memory
  - Buffers for stimuli received through the senses:
    - iconic – visual stimuli
    - echoic – aural stimuli
    - haptic – touch
  - Constantly overwritten as new information arrives.

Human - STM

- Short-Term Memory (STM)
  - "Scratch-pad" for temporary recall of information
    - Example: Mental calculations, reading.

    - Quick access: 70ms
    - Quick decay: 200ms
    - Interference causes faster decay
    - Limited capacity: 7 $\pm$ 2 information blocks (Miller's law).
    - Desire to complete and close tasks held in the STM

Human - STM

- Example:

7561093

Human - STM

- Example:

?

Human - STM

- Example:

36B789C563

Human - STM

- Example:

?

Human - STM

- Example:

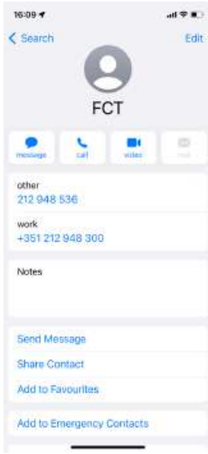
643 71B 83M6

Human - STM

- Example:

?

Human - STM



Human - STM

- Example:

WAU HTP NYD KSD YHB

IBM BMW FBI URL ATM

ABC DEF GHI JKL MNO

Human - Memory

- Can you remember a 50-digit number?
  (after seeing it for 1 second)

11111111111111111111111111111111111111111111111111

Human - LTM

- Long-Term Memory
  - Knowledge repository
    - Slow access time: 1/10 second
    - Little decay (if any)
    - Huge capacity (or unlimited)
  - Two types:
    - Episodic: memory of serial events.
      - Ex: remember the events that took place in a certain moment of our lives.
    - Semantic: structured record of facts, concepts and skills. Represents relationships between information.
      - Ex: if Snoopy is a dog => Snoopy has 4 legs.
    - Semantic LTM derived from episodic LTM

Human - LTM

- Processing in the LTM
  - Information storage
    - STM $\rightarrow$ LTM by rehearsal
    - Studies show:
      - Total time hypothesis - The amount of information learned is proportional to the amount of time spent learning.
      - Distribution of practice effect - Learning time is most effective if it is distributed over time.
      - Structure, meaning and familiarity make information easier to remember.

Human - LTM

- Processing in the LTM
  - Forgetting
    - Information is gradually and slowly lost.
    - LTM is selective and influenced by emotions
      - We tend to remember highly emotive events than mundane ones.
      - “Good old days”
    - Apparently, new information replaces the old one (retroactive inhibition), but sometimes old memory interferes with new information (proactive inhibition).
    - Do we forgot information or we just are not able to retrieve it?
      - Tip of the tongue experience
      - Recognition

Human - LTM

- Processing in the LTM
  - Information retrieval
    - Recall (relembrar)
      - Information is reproduced from memory. Cues can be helpful (categories, images, …)
    - Recognition (reconhecer)
      - The presentation of the information provides the knowledge that the information has been seen before. Easier than the recall process – the information is the cue.
    - Examples: Colleagues from the 4th grade, quiz shows.

Human - Memory

- You will see 10 3-character strings
- One at a time
- Try to remember them
- The order is not important
- Can´t write them down before I say so

Human - Memory

WAT
HEP
CAX
NOF
TEH
DOK
RIJ
ZIB
BAL
MEQ

Human - Memory

Write down the strings you remember, now!

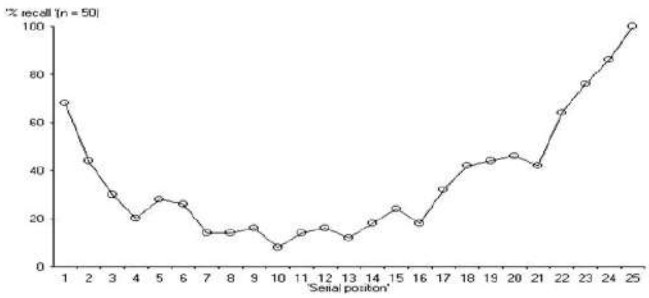
Human - Memory

WAT
HEP
CAX
NOF
TEH
DOK
RIJ
ZIB
BAL
MEQ

Human - Memory

- Primacy and recency effects

  Typically, words at the start of the list and especially those at the end tended to be recalled most often.



Murdock, 1962

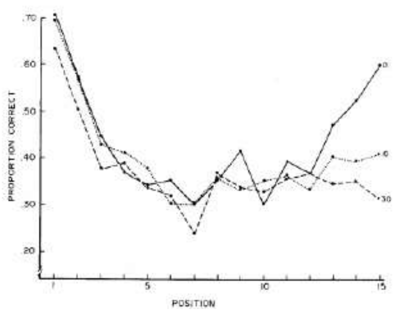
Human - Memory

- Same exercise, but…
- Wait 30 seconds (count down from 30) before recall (writing down the strings).

Human - Memory

- Primacy and recency effects

Delaying recall by 30 seconds prevented the recency effect.

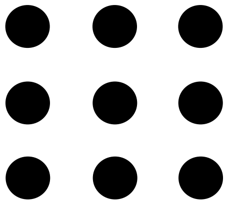


Glanzer and Cunitz, 1966

Human - Thinking

- Problem solving

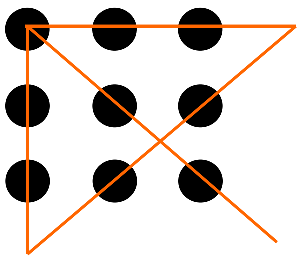
📖 Conceptual Blockbusting, James L. Adams, Basic Books, New York, 2001



Human - Thinking

- Problem solving

📖 Conceptual Blockbusting, James L. Adams, Basic Books, New York, 2001



Human – Individual differences

- Individual differences
  - In interface design we should consider individual differences
  - Three main types of differences
    - Long term: sex, physical and intellectual capabilities
    - Short term: stress, fatigue, …
    - Changes: age, idiosyncrasies…
  - Be aware if a design decision may exclude part of the target users population.
  - In the same group of target users significant differences can be noticed.
  - **The users should not be forced to work on their perceptual and cognitive limits**. They should feel comfortable in using the systems.

Human - Emotions

- Emotions
  - The biological response to physical stimulus is called affect.
  - Affect influences how we respond to situations
    - Positive emotions – creative thinking, complex problem solving
    - Negative emotions – restrict reasoning.

“Negative affect can make it harder to do even easy tasks; positive affect can make it easier to do difficult tasks.”

Donald Norman, Emotional Design

- Build interfaces that promote positive responses.

References

- Adams, J. L., Conceptual Blockbusting, Basic Books, New York, 2001
- Dix, Alan, Finlay, Janet, Abowd, Gregory, Beale, Russel. Human-Computer Interaction (3rd Edition), Prentice Hall, 2004.
- Hick, W. E., On the Rate of Gain of Information. Quaterly Journal of Experimental Psychology, 4, 1952, pp. 11-26.
- Norman, Donald. The Design of Everyday Things. MIT Press, 1998.
- Norman, Donald. Emotional Design, Basic Books, New York, 2004.
- Adams, J. L., Conceptual Blockbusting. Basic Books, New York, 2001.
- Card, S.K., Moran, T. P. and Newell, A., The Psychology of Human Computer Interaction. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1983.