DESIGNING THE USER INTERFACE

USER INTERFACE

COURSE 2020/2021

INTERACTION DESIGN

In this topic, we will analyze in detail the following concepts related to the design process, already introduced in Topic 2:

- The process of interaction design: Life-cycles
- What are the requirements?
- Types of requirements
- Data gathering
- Interpretation and data analysis
- Conceptual design

INTERACTION DESIGN

The process of interaction design: Life-cycles

What are the requirements?

Types of requirements

Data gathering

Interpretation and data analysis

Conceptual design

LIFE-CYCLE MODEL(S)

Interaction design is based on the premise that design should emerge from:

- The users' tasks
- The users' goals, and
- The environment or context in which the interaction takes place.

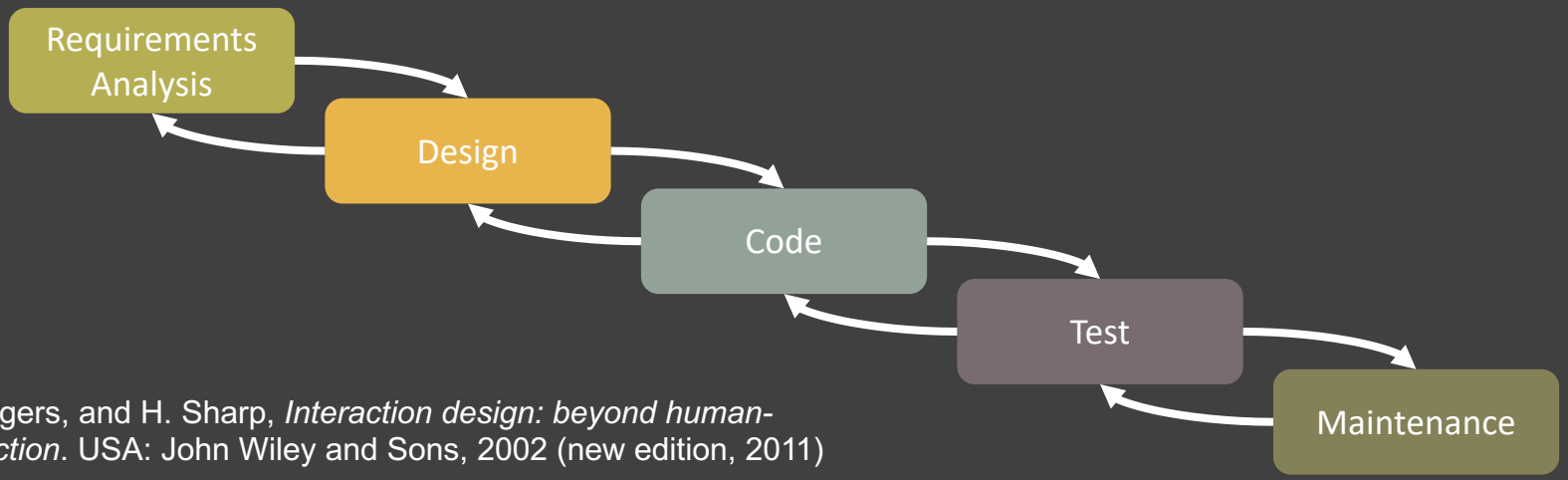
The central focus of interaction design is to acquire a thorough understanding of the people who will use the proposed system

Attempts to standardize this process include the life-cycle models used in Software Engineering

LIFE-CYCLE MODEL(S)

Waterfall model

- Classic and one of the first ones generally used in the software industry
- It emphasizes a linear, sequential approach to software development
- It seems quite logical: one must first consider the requirements of the system and then have a basic design before starting the implementation and testing phase



J. Preece, Y. Rogers, and H. Sharp, *Interaction design: beyond human-computer interaction*. USA: John Wiley and Sons, 2002 (new edition, 2011)

LIFE-CYCLE MODEL(S)

The waterfall model is based on the notion that the requirements that have been defined in the initial phase of development cannot be changed during the following phases

The waterfall model is not a user-centred model in that it does not formally take the user into consideration

- Interaction design is an iterative process. Why?

Over time, most waterfall models include some limited iteration between the intermediate stages of development

Agile Development

Working software
VS
Comprehensive documentation

Individuals and interactions
VS
Processes and tools

Responding to change
VS
Following a plan

Customer collaboration
VS
Contract negotiation

http://agilemanifesto.org/

Agile Design

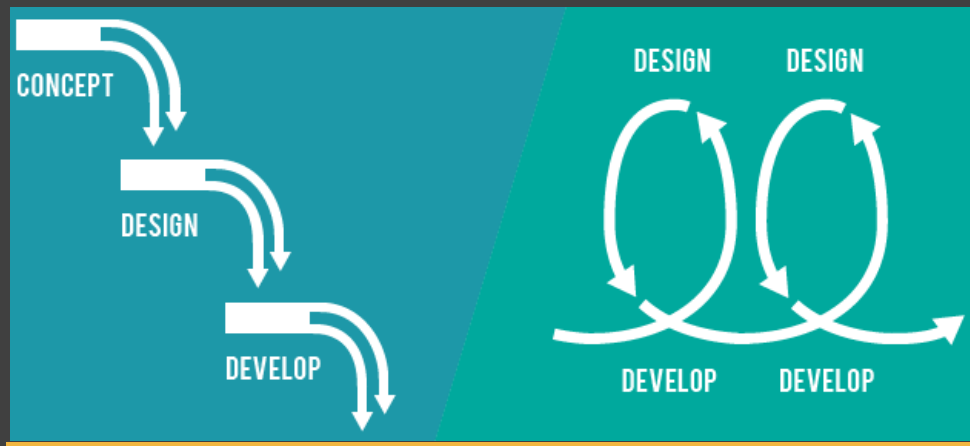
Frequent feedback from the customers to be applied to the design of the product

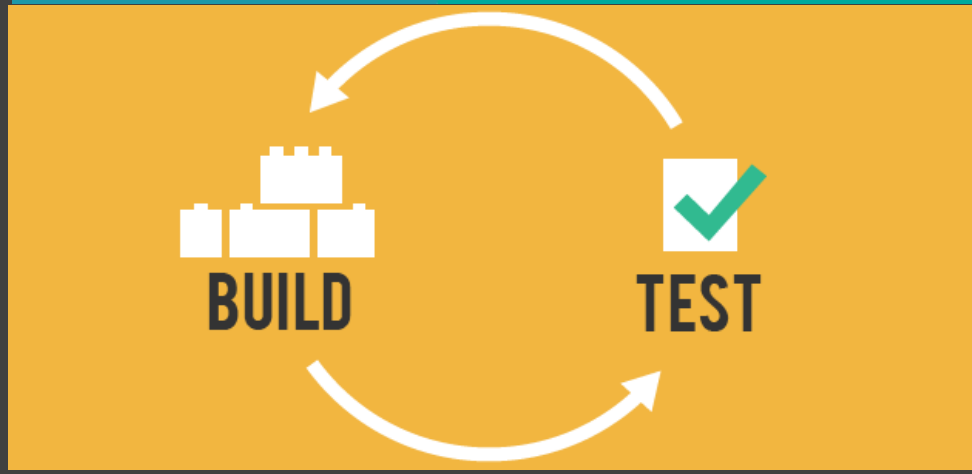
Make the customer a member of the team

Frequent check-in within the design team

Frequent testing

“Problem solving” mentality





http://designshack.net/articles/business-articles/understanding-agile-design-and-why-its-important/

INTERACTION DESIGN

The process of interaction design: Life-cycles

What are the requirements?

Types of requirements

Data gathering

Interpretation and data analysis

Conceptual design

REQUIREMENTS

What are the requirements in interaction design?

Terms related to requirements:

- Gathering / Collection of requirements
  - To identify and capture requirements from the environment
- Analysis of requirements
  - To investigate or analyse an initial set of requirements
- Requirements engineering
  - Software engineering process which covers all of the activities involved in discovering, documenting and maintaining a set of requirements for a computer-based system. It is an iterative collection and analysis process

REQUIREMENTS

What are the requirements in interaction design?

A requirement is a singular documented need that a particular product or service must perform (functional requirement) or be (non-functional requirement)

Examples:

- The maximum amount of time to download the home page of our site should not exceed 5 seconds (functional)
- Users must find the application attractive (non-functional)

REQUIREMENTS

There are different types of requirements

Some recommendations:

- Clear: we use them to develop software and design user interfaces. If they are difficult to understand, or vague, they won't be useful
- Easy to verify: requirements should be described in such a way that evaluations of the system allow us to confirm that the system meets them

INTERACTION DESIGN

The process of interaction design: Life-cycles

What are the requirements?

Types of requirements

Data gathering

Interpretation and data analysis

Conceptual design

TYPES OF REQUIREMENTS

1) Functional: what the system should do

Examples

- Different format styles in word processor
- Change of temperature in a reactor

2) Non-functional: restrictions related to the system and its development. In other words, criteria that can be used to judge the operation of a system, rather than specific behaviors

- The system must work on different operating systems (e.g. Windows and Mac)
- Must be able to connect through the IEEE 488 bus

TYPES OF REQUIREMENTS

When physical devices are considered there are characteristics such as shape, weight, size, etc. that must be considered.

→ Are these characteristics functional or non-functional?

Non-functional might impact on functional

- For example, the size of a mobile device
- Broadly, functional requirements define what a system is supposed to do whereas non-functional requirements define how a system is supposed to be

TYPES OF REQUIREMENTS

A more exhaustive list of requirements. Four categories of requirements:

- Data: Capture the type, volatility, size/amount, persistence, precision and value of the required data.
- Environment: Refer to the circumstances in which the interactive product is expected to operate. Physical (e.g. light, noise), social (e.g. collaboration), organizational and technical environment (e.g. compatibility)
- User: Capture the characteristics of the intended user group: talent and skills; casual or frequent; expert or novice. The result is the user profile
- Usability: Capture the usability goals and associated measures: effectiveness, efficiency, safety, usefulness and learning.

TYPES OF REQUIREMENTS

EXERCISE:

Suggest one key functional, data, environmental, user and usability requirement for the following scenario:

- A system to be used in a university's self-service cafeteria that allows users to pay for their food using a credit system

INTERACTION DESIGN

The process of interaction design: Life-cycles

What are the requirements?

Types of requirements

Data gathering

Interpretation and data analysis

Conceptual design

DATA GATHERING

We aim to collect sufficient, relevant and appropriate data to define a stable set of requirements.

- Sufficient – lots of data to be gathered, can we gather all?
- Relevant – depending on our system

If we already have a set of requirements, we can gather or collect more data in order to:

- Deepen and widen the requirements
- Clarify the requirements which are vague
- Confirm the requirements gathered some time ago

DATA GATHERING

We often need to know

- How tasks are currently conducted
- What are the objectives
- Where tasks are conducted (the context)
- Why things are the way they are

Different techniques – which ones do you know?

- Questionnaires, Interviews, Focus groups and workshops, Observation, Document review and Analysis of logs
- Non-exclusive. Why?
  - Triangulation - See the same phenomenon from (as many) different points of view (as possible)

QUESTIONNAIRES IN HCI

QUIS (Questionnaire for User Interface Satisfaction)

- Assess subjective satisfaction with specific aspects of HCI: screen factors, on-line tutorials, multimedia, learning factors, etc.
- Purposes: Guide in the design and redesign of systems; Assessing potential areas of system improvement; conducting comparative evaluations.

CSUQ (Computer System Usability Questionnaire)

- 19 questions that measure user satisfaction regarding the usability of the system

QUESTIONNAIRES IN HCI

PUEU (Perceived Usefulness and Ease of Use)

- Developed to support the model of technology acceptance: the user acceptance is determined by the perceived usefulness and perceived ease of use

NAU (Nielsen's Attributes of Usability)

- Learnability: systems should be easy to learn
- Efficiency: systems should be efficient to use
- Memorability: systems should be easy to remember, without having to learn everything all over again
- Errors: the system should have a low error rate
- Satisfaction: the system should be pleasant to use

QUESTIONNAIRES IN HCI

ASQ (After Scenario Questionnaire)

- Addresses three important components of user satisfaction with system usability: ease of task completion, time to complete a task, and adequacy of support information

PUTQ (Purdue Usability Testing Questionnaire)

- Evaluates 8 factors relevant in HCI: compatibility, consistency, flexibility, learnability, minimal action, minimal memory load, perceptual limitation, and user guidance

QUESTIONNAIRES IN HCI

SUMI (Software Usability Measurement Inventory)

- SUMI is a consistent method for assessing the quality of use of a software product or prototype, and can assist with the detection of usability flaws before a product is shipped (http://sumi.ucc.ie/)

SUS (System Usability Scale)

- SUS is a simple, ten-item attitude Likert scale giving a global view of subjective assessments of usability.
- http://en.wikipedia.org/wiki/System_usability_scale

QUESTIONNAIRES IN HCI

SUS (System Usability Scale) – Standard Version

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

Where each item is scored with a Likert Scale from Strongly Agree (5) to Strongly Disagree (1).

QUESTIONNAIRES IN HCI

How to apply the SUS test.

1. Answer to the previous 10 questions with a value from 1 to 5, based on their level of agreement.
2. For each of the odd numbered questions, subtract 1 from the score.
3. For each of the even numbered questions, subtract their value from 5.
4. Take these new values which you have found, and add up the total score. Then multiply this by 2.5.
5. The result is a score between 0 and 100 and it can be interpreted in the following way:
   - 80.3 or higher is an A. People love the website and will recommend it to their friends
   - 68 or thereabouts gets a C. The website is OK but it could be improved
   - 51 or under gets you a F. The usability of the website should be fit.

QUESTIONNAIRES IN HCI

Technology Acceptance Model (TAM) - a very different perspective, more oriented to the developed technology, and one of the most used nowadays.

- Performance expectancy
- Effort expectancy
- Attitude toward using technology
- Social influence
- Facilitating conditions
- Self-efficacy
- Anxiety
- Intention to use the system

V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, 2003. USER ACCEPTANCE OF INFORMATION TECHNOLOGY: TOWARD A UNIFIED VIEW, MIS Quarterly, vol. 27, no. 3, pp. 425-478

QUESTIONNAIRE DESIGN

Question wording and layout are very important components of effective questionnaires

- If respondents find questions difficult to understand or if questions are either ambiguous or irrelevant for the research purposes, questionnaires will not provide useful data
  - E.g. Is the interface easy to use? Yes - No

Validity = refers to whether questionnaires measure what they are expected to measure,

Reliability = refers to questionnaire's consistency

QUESTIONNAIRE DESIGN

The most difficult part of designing a questionnaire is to design the pilots

- Pilots are preliminary versions of questionnaires
- Pilots are evaluated with a sample of users and allow researchers to design the final questionnaire by identifying problems and testing solutions in previous versions

QUESTIONNAIRE DESIGN

Open or free questions are not followed by any kind of choice.

Instead, respondents are required to write their answer. Open questions allow users to express their points of view and are easy to ask.

Nevertheless, open questions are quite difficult to analyze, due to the ambiguity of natural language, and demand more effort from respondents (e.g. to think of the answer)

QUESTIONNAIRE DESIGN

Example: How would you improve the system?

- User: Which part, the user interfaces, or the technology, or both?
- Researcher: I have got a 5-line answer from all the questionnaires. I have administered 20 questionnaires, so I have to analyze a 100-line document

QUESTIONNAIRE DESIGN

Closed questions are those in which the respondents are offered a choice of alternative answers

- Check-boxes and ranges: typically used for answers like Yes or No and ages (e.g.; < 3 months; between 3 and 6 months; > 6 months)
- Likert scales: typically used for measuring user opinions (e.g. strongly agree; agree; strongly disagree)
- Semantic differential scales: for measuring users' attitudes (e.g. easy 1- 2-3-4-5 difficult)

INTERVIEWS

Interviews have long been used in many disciplines. Interviewing is an interactive exchange of asking questions and getting answers.

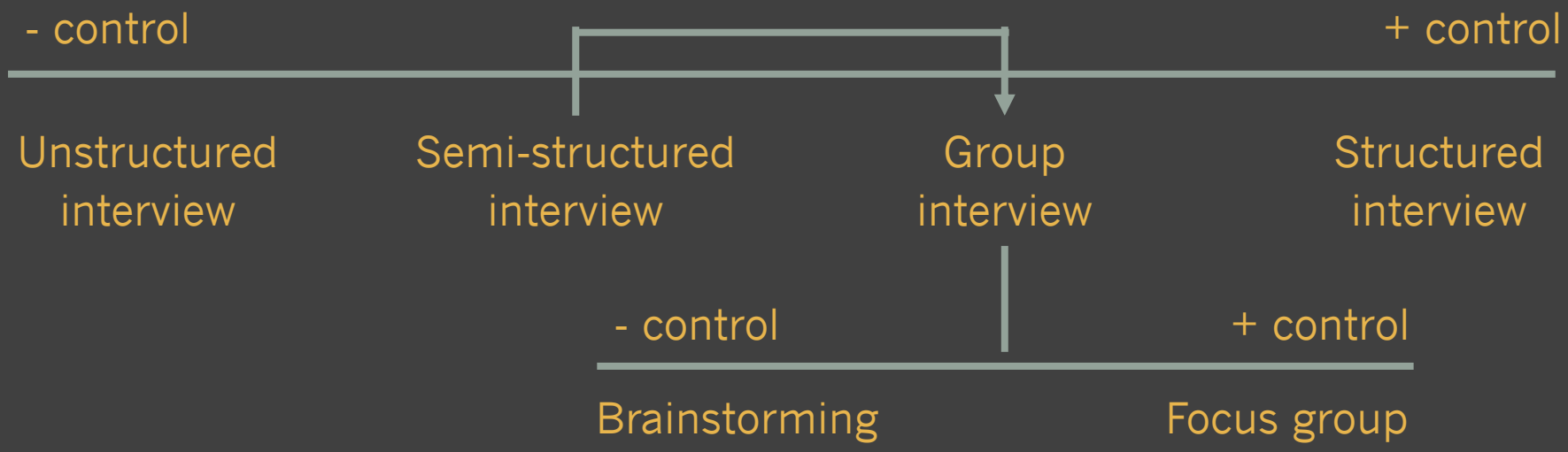
Interviews only reveal part of the information.

Interviews are time- consuming and expensive.

Interviewing has to cope with the careful preparation of materials, the ambiguity of natural language, the variability of people and the costs of travelling to respondents' context.

INTERVIEWS

The control the interviewer takes over the interview varies according to the goal of the research. It can vary ranging from discovering topics to gathering specific responses.



INTERVIEWS

Structured Interviews

- A small room is used
- Pre-determined list of questions and answers (i.e. a questionnaire). The questionnaire is read aloud and the role of the user is to select answers. The interviewer plays a neutral role and takes a lot of control over the interview
- Preparatory work is essential in order to capture the specific data in a precise manner. The interviews can easily be repeated and the findings analyzed (standard study)
- They do not capitalize on the social interaction established between interviewers and interviewees. Also included phone interviews and intercept interviews in malls and parks.

INTERVIEWS

Unstructured interviews

- They resemble conversations focused on a topic
- The used script solely contains the main topics to be addressed during the interview. The topics are generally asked by means of open questions, unlike structured interviews.
- Other issues might come up during the interview because of the unstructured conversation style. They are usually added to the list and addressed in the interview.
- Capture and generate rich and contextual data, even if their analysis is time-consuming, because we are dealing with free conversations.

INTERVIEWS

Semi-structured interviews

- They combine features of structured and unstructured interviews.
- The order of questions in the script is not strict, unlike structured interviews. Other topics, which might be brought up by interviewees, might be added to the list and therefore addressed in the interview.
- Open questions are usually asked in semi-structured interviews. Nevertheless, these questions tend to be a bit more specific than the questions asked in unstructured interviews.
- The interviewers have a more defined idea about those topics or information which needs to be gathered.

INTERVIEWS

Group interviews (aka focus groups)

- They consist in the systematic questioning of several individuals simultaneously in a formal or informal setting.
- Taking control over a group is more difficult than controlling a person.
- The interviewer's control can vary depending on the type of group interview. For instance, focus groups seem to be the most standard group interview.
- There are other types of group interviews, such as *brainstorming* interviews, which require less control than focus groups

OBSERVATION

Participant observation

- Observe activities and participate in them
- First-hand observations: ‘what people say they do (i.e. in questionnaires and interviews) and what they really do tends not to be the same’
- Time-consuming (in anthropology, years; in sociology, years and months)
- Lots of (rich) data

Observation can be overt and covert

MORE TECHNIQUES

Document review

- Rules and procedures
- Quantitative
- Users are not really involved
- Daily routine versus procedures

Analysis of Logs

- How the system works (metrics)
- Quantitative
- “Less” involvement of the user
- Opinions are not taken into account

WRAPPING-UP

What data do we need to design our system?

- At the start of the project, do we need interviews or questionnaires?

What resources do we have?

- If we have two weeks, what type of questionnaires / interviews do we conduct? Should we carry out observations?

Where are the users?

How easy or difficult is it to get hold of them?

WRAPPING-UP

Two aspects to take into account in the selection of the data collection technique:

- The nature of the data (amount of time and level of detail, analysis of the data)
- The task to be studied (Is it sequential or tasks overlap? Does the task involve complex or simple information? Is the task going to be done by an expert or a layman?)

WRAPPING-UP

Guide for collecting data

- Focus on identifying the needs of users
- Involve all types of user groups
- Involve more than one user of those groups
- Use a combination of techniques
- Conduct a pilot test of data collection
- Commitment between the idyllic situation and the restrictions of reality
- Practical data collection

EXERCISE

You are about to start a project

You have resources to gather requirements during a relatively long period of time

You have 3-4 months before the coding of the system starts

You are about to design an e-commerce website targeted at young Spanish people

What data gathering techniques will you use to know what the system must do and how it should be?

SOLUTION

Can you interview all your potential end-users?

- Will focus groups be a suitable technique?

How useful questionnaires could be?

- What type of questionnaire would you design, online or paper-based questionnaires?
- Why?

Will you conduct documentation review? Why?

SOLUTION

It’s not possible to interview every user nor define user groups.

Questionnaires and focus groups are the most appropriate techniques.

- Please, write your name and age. Name?
- Is the interface easy to learn? Yes/No (useful?)

Identify and evaluate similar websites for improving the product to design.

INTERACTION DESIGN

The process of interaction design: Life-cycles

What are the requirements?

Types of requirements

Data gathering

Interpretation and data analysis

Conceptual design

Interpretation & Data Analysis

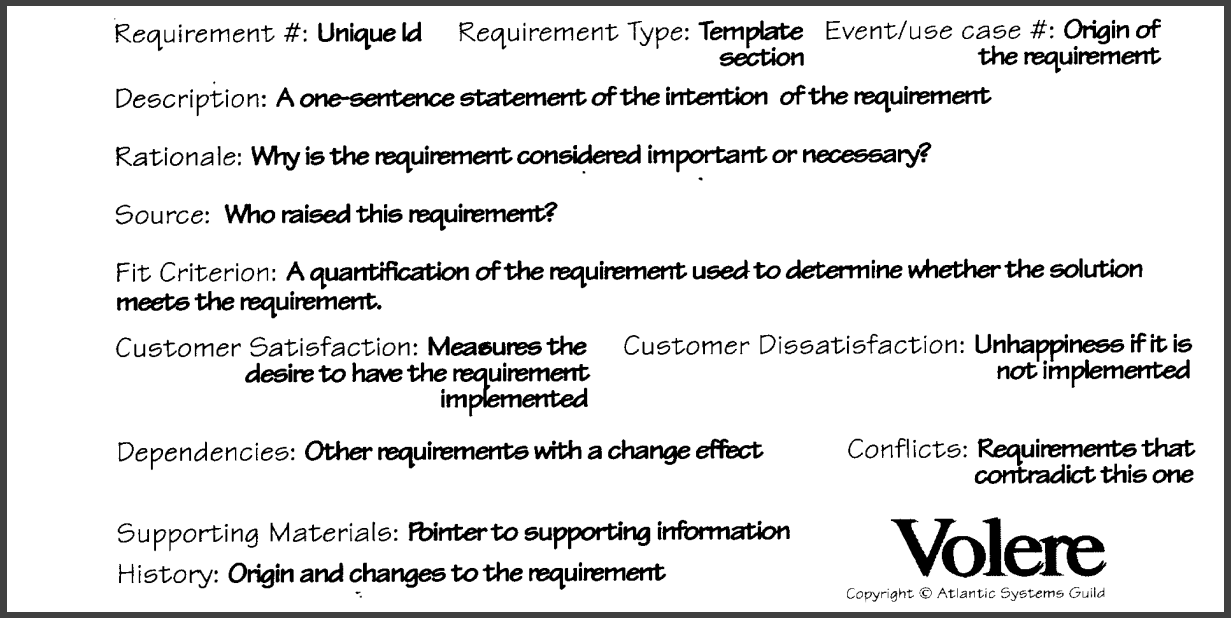
We have looked at ways to gather information and gain an understanding about the workflow that will be incorporated into the design.

How to organise this information and describe it in such a way that is useful during the design phase?

- The purpose of the interpretation phase is to structure and turn these data into requirement specification templates / documents to help us manage the information.

Interpretation & Data Analysis

Requirements specification: documents / diagrams with information regarding identifier of the requirement, type, description, rationale, source, fit/adaptation criteria, user satisfaction, relationship with other requirements, history, …



A guide to writing appropriate requirements specifications: https://www.volere.org/templates/volere-requirements-specification-template/ (steps 9,10,11)

TASK ANALYSIS

Task analysis:

- A way of documenting how people perform tasks
- A way of understanding goals and objectives

Task analysis techniques used throughout the iterative process of design to gain a deep understanding of the requirements

TASK ANALYSIS

The analysis has to do with the investigation of the different aspects of the system according to the established requirements…

- functional (eg, state diagrams, data flow diagrams, ...), data (eg entity-relation), class and sequence diagrams, ...

…and the techniques to understand the goals and objectives of the users, and to understand in depth the requirements

- Description of tasks - describe the tasks by which the system will be accepted: scenarios and use cases
- Task analysis - the information collected establishes the basis of the analysis practices to evaluate the defined tasks, such as HTA and GOMS

TASK ANALYSIS: SCENARIO

Scenario

- A description in plain English of a typical task end-users may wish to accomplish with the proposed design.
- It describes the basic goal, the conditions that exist at the beginning of the task, the activities in which the end-user or end-users will engage, and the outcomes of those activities.
- It allows the exploration and discussion of contexts, needs and requirements.
- Usually the first scenarios are built by the users, and can describe what it is like or how it is expected to be.
- It is based on a narrative and an objective and incorporates: Actors, Activities and Objects.

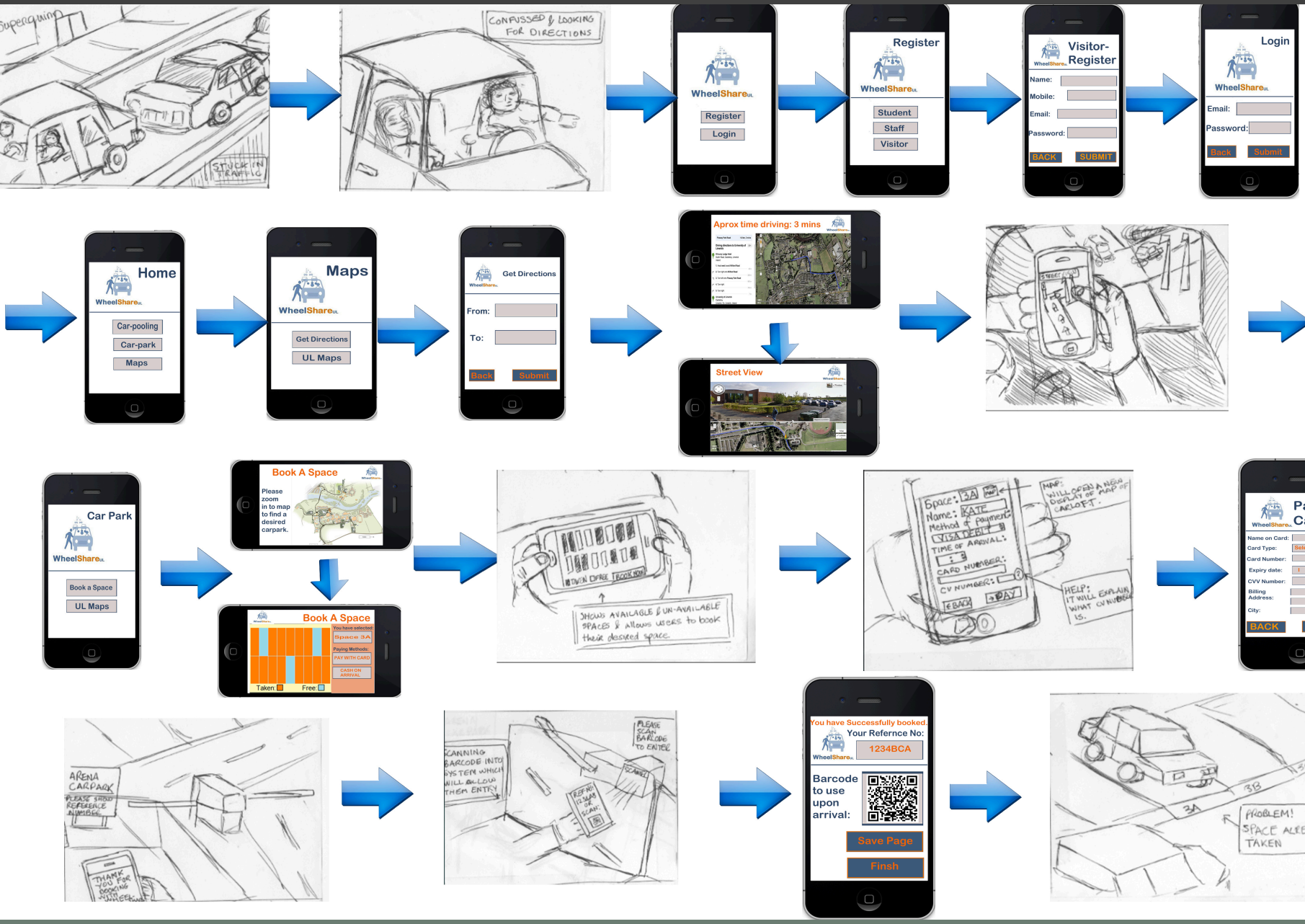
TASK ANALYSIS: SCENARIO

Type of representations for scenarios

- Narrative: a complete story of the interaction made with the existing design or with a new design
- Flow diagram: a graphic representation of the series of actions and decisions taken from the narrative
- Procedural texts: a step-by-step description of user actions and system responses
- Storyboard: a graphic narration of a story in consecutive pictures. This concept comes from the cinematographic design and is often used for the realization of interaction scenarios that can later be evaluated with different techniques.

TASK ANALYSIS: SCENARIO

Storyboard of an App to reserve a parking place (see next slide…)


CONFUSSED & LOOKING FOR DIRECTIONS
STUCK IN TRAFFIC
WheelShare
Register
Login
Student
Staff
Visitor
Visitor-Register
Name:
Mobile:
Email:
Password:
BACK
SUBMIT
Email:
Password:
Back
Submit
Home
Car-pooling
Car-park
Maps
Maps
Get Directions
UL Maps
Get Directions
From:
To:
Back
Submit
Aprox time driving: 3 mins
Street View
Car Park
Book a Space
UL Maps
Book A Space
Please zoom in to map to find a desired carpark.
Book A Space
You have selected
Space 3A
Paying Methods:
PAY WITH CARD
CASH ON ARRIVAL
Taken
Free
SHOWS AVAILABLE & UN-AVAILABLE SPACES & Allows users to book their desired space
Space: 3A
Name: KATE
Method of payment
VISA DEBIT
TIME OF ARRIVAL:
CARD NUMBER:
CV NUMBER:
BACK
PAY
HELP: IT WILL EXPLAIN WHAT CV NUMBER IS.
Pay with Card
Name on Card:
Card Type:
Select an option
Card Number:
Expiry date:
CVV Number:
Billing Address:
City:
BACK
SUBMIT
ARENA CARPARK
PLEASE SHOW REFERENCE NUMBER
THANK YOU FOR BOOKING WITH WHEELSHARE
SCANNING BARCODE INTO SYSTEM WHICH WILL ALLOW THEM ENTRY
PLEASE SCAN BARCODE TO ENTER
SCANNER
You have Successfully booked.
Your Refernce No:
1234BCA
Barcode to use upon arrival:
Save Page
Finsh
3B
3A
PROBLEM! SPACE ALREADY TAKEN

TASK ANALYSIS: SCENARIO

Two general types of scenarios:

- Task-based scenarios: state only what the user wants to do. Scenarios at this level do not include any information on how the user goes about completing the scenario
  - Example: You are traveling to Dundee for your job next week and you want to check on the amount you can be reimbursed for meals and other expenses
  - Students in order to open their account in the department labs have to provide their ID number and their personal information through a form. The system needs to check that all the required data have been introduced, otherwise the request can not be processed.

- Elaborated scenarios: give the development team a deeper understanding of the users (example in next slide)

TASK ANALYSIS: SCENARIO

Example: Mr. and Mrs. Macomb are retired schoolteachers who are now in their 70's. Their Social Security checks are an important part of their income. They've just sold their big house and moved to a small apartment. They know that one of the many chores they need to do now is to tell the Social Security Administration that they have moved. They don't know where the nearest Social Security office is and it's getting harder for them to do a lot of walking or driving, so they would like to do this on the computer if it is easy and safe enough. However, they are somewhat nervous about doing a task like this by computer. They never used computers in their jobs; but their son, Steve, gave them a computer last year, set it up for them, and showed them how to use email and how to go to Web sites. They have never been to the Social Security Administration's Web site, so they don't know how it is organized. Also, they are reluctant to give out personal information on Web sites, so they want to know how safe it is to tell the agency about their new address this way.

http://www.usability.gov/methods/analyze_current/scenarios.html

TASK ANALYSIS: SCENARIO

What things do we need to consider in the previous scenario?

TASK ANALYSIS: SCENARIO

Example: Mr. and Mrs. Macomb are retired schoolteachers who are now in their 70's. Their Social Security checks are an important part of their income. They've just sold their big house and moved to a small apartment. They know that one of the many chores they need to do now is to tell the Social Security Administration that they have moved. They don't know where the nearest Social Security office is and it's getting harder for them to do a lot of walking or driving, so they would like to do this on the computer if it is easy and safe enough. However, they are somewhat nervous about doing a task like this by computer. They never used computers in their jobs; but their son, Steve, gave them a computer last year, set it up for them, and showed them how to use email and how to go to Web sites. They have never been to the Social Security Administration's Web site, so they don't know how it is organized. Also, they are reluctant to give out personal information on Web sites, so they want to know how safe it is to tell the agency about their new address this way.

http://www.usability.gov/methods/analyze_current/scenarios.html

TASK ANALYSIS: SCENARIO

What things do we need to consider in the previous scenario?

- Easy and safe
- Lack of experience with similar sites
- Their son will probably give them a hand
- Privacy (personal information)
- Older people
- …

TASK ANALYSIS: USE CASES

Use cases

Represent a formal, structured approach to interpreting work flows and processes

- They are designed to describe a particular goal and explore the interaction between users and the system
- User perspective → Although it focuses on the interaction between the user (actor) and the software system, it is seen from the perspective of the user, not the system

TASK ANALYSIS: USE CASES

Two main components

- Actors: anything that has contact with or is affected by the system (E.g. humans and software)
- Use case: task or goal the actor is engaged in

There is a scenario in the use cases that indicates the path through a specific set of conditions. There are generally different paths:

- Basic path: without any diversions – Main use case
- Alternate path: paths to take depending on the situation

TASK ANALYSIS: USE CASES

Example of use case - Mr. and Mrs. Macomb

- Basic path
  - Mr. and Mrs. Macomb enter the URL in the web browser
  - The system asks for address
  - Mr. and Mrs. Macomb trust the system and enter address
  - The system shows confirmation message
- Alternate path
  - Mr. and Mrs. Macomb do not trust the system
  - They give a phone call to Steve
  - Steve introduces the address remotely for them

TASK ANALYSIS: USE CASES

Example of use case – Library catalogue service

- Basic path
  - The user starts a session
  - The system asks for ID and password
  - The user enters the ID and password
  - The system checks that they are correct
  - The system shows the search screen
  - …
- Alternate path
  - 4. If the ID does not exist or the password does not match the ID:
    - 4.1 the system displays an error
    - 4.2 the system asks again the user for ID and password

TASK ANALYSIS

There are different techniques to analyze tasks

- Hierarchical Task Analysis (HTA) - allow dividing a task into sub-tasks recursively to understand how to carry them out
- Predictive models - *a priori* models used to define an approximation of the actions that users will execute before involving the users themselves in real tests.
  - Examples: Model Human Processor (MHP), Keyboard Level Model (KLM), and Goal Operators Methods and Selection rules (GOMS)
- Descriptive models - framework in which the user interacts with the system.
  - Example: Three-State Model

HIERARCHICAL TASK ANALYSIS

HTA was originally designed to identify training needs.

HTA focuses on the physical and observable actions that are performed, and includes looking at actions that are not related to the software or the interaction device.

It involves breaking a task into subtasks.

Process:

- The starting point is a user goal, then examined to identify the main tasks for achieving it.
- If needed, these tasks are subdivided into subtasks and then into subtasks and so on. This is an iterative process that will continue until reaching the required level of granularity.
- Subtasks are then grouped together in a plan. Each plan specifies how the tasks might be performed in an actual situation.

HIERARCHICAL TASK ANALYSIS

Consider the library catalogue service and the task of borrowing a book

This task can be decomposed into other tasks, such as accessing the library catalogue, searching by name, title, subject…going to the correct shelf and finally taking the book to the check-out counter

This set of tasks and subtasks might be performed in a different order depending on how much is known about the book, and how familiar the user might be with the library and the book's likely location

HIERARCHICAL TASK ANALYSIS

HTA:

0. In order to borrow a book from the library
  - 1. Go to the library
  - 2. Find the required book
    - 2.1 Access library catalog
    - 2.2 Access the search screen
    - 2.3 Enter search criteria
    - 2.4 Identify required book
    - 2.5 Note location
  - 3. Go to correct shelf and retrieve book
  - 4. Take book to check-out counter
  - Plan 0: do 1-3-4. If book is not on the shelf expected, do 2-3-4
  - Plan 2: do 2.1-2.4-2.5. If book not identified do 2.2-2.3-2.4-2.5

HIERARCHICAL TASK ANALYSIS

Plans control /show the flow

- Plan 0 assumes the book is on the correct shelf
- Plan 2 shows different ways of looking for a book

Indentation shows the hierarchical relationship between tasks and subtasks

Note how the numbering works for the tasks analysis: the number of the plan corresponds to the number of the step to which the plan relates

HIERARCHICAL TASK ANALYSIS

An alternative expression of an HTA is a graphical box-and-line notation

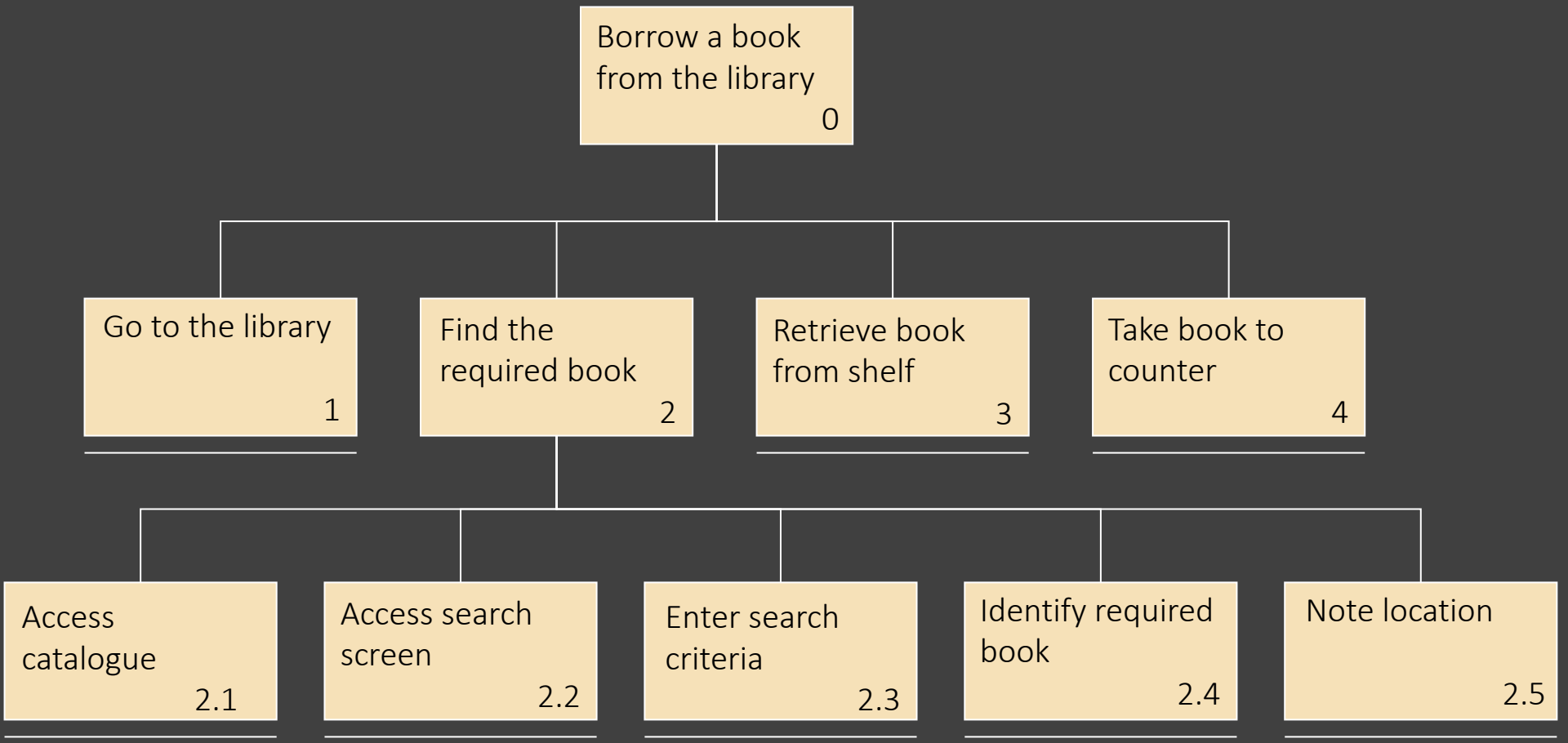
- The subtasks are represented by named boxes with identifying numbers
- The hierarchical relationship between tasks is shown using a vertical line
- If a task is not decomposed any further then a thick horizontal line is drawn underneath the corresponding box

Plans are also shown in this graphical form

Plan 0: do 1-3-4. If book is not on the shelf expected, do 2-3-4

Plan 2: do 2.1-2.4-2.5. If book not identified do 2.2-2.3-2.4-2.5

HIERARCHICAL TASK ANALYSIS



EXERCISE

Carry out an HTA (Hierarchical Task Analysis) for arranging a meeting in a calendar application shared by several members of a company working at different departments.

EXAM QUESTION

(January 2013) Carry out an HTA (Hierarchical Task Analysis) of the following task: to get your coffee (espresso, white coffee, Americano, cappuccino…) from a vending machine (2 points).

PREDICTIVE MODELS - MHP

Model Human Processor (MHP) is used to make predictions on how a user will execute the tasks.

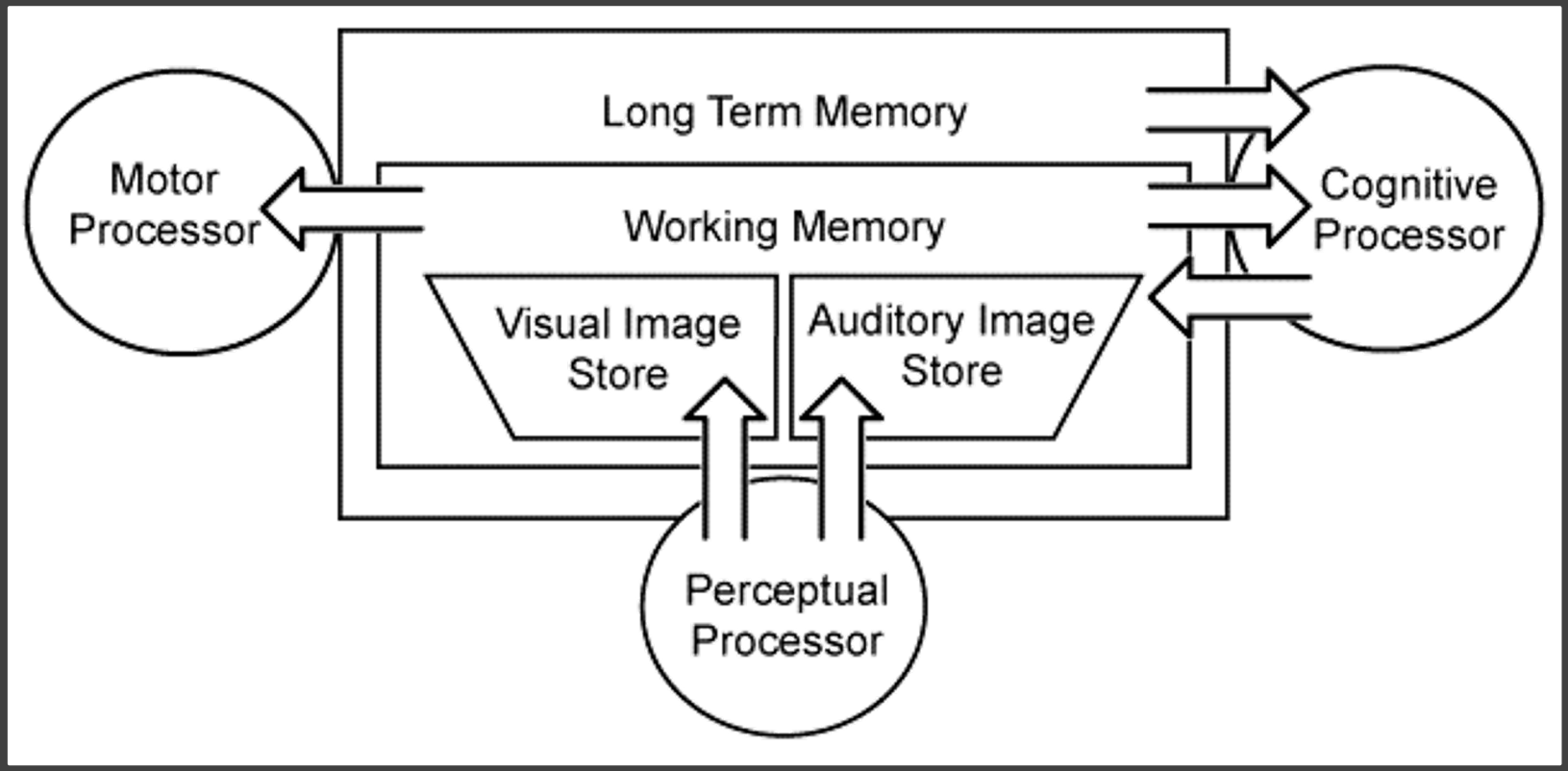
It is formed by a group of memory stores and processors that function following some operation principles.

Perception system

- Sensors: eyes, ears
- Buffers: Visual memory store (VIS), Auditory memory store (AIS)
- Cognitive system: Working memory (WM) or short-term memory, Long-term memory (LTM)
- Motor system: arm-hand-finger system, head-eye system

PREDICTIVE MODELS - MHP

Model Human Processor (MHP)



PREDICTIVE MODELS - GOMS

GOMS (goals, operators, methods and selection rules) – it aims to model the knowledge and cognitive processes generated when the user interacts with the system.

- Goals – "memory points" that enable the user to evaluate his/her progress.
- Operators – the lowest level in the model, e.g. printing a document requires issuing a print command, which may involve clicking the printer icon, an operator.
- Methods – e.g. printing a document can be achieved in different ways (shortcut, icons…)
- Selection rules – e.g. if the user is involved in intensive typing, shortcuts might be better; but if he/she is heavily mouse oriented, an icon might be the best option.

PREDICTIVE MODELS - GOMS

- GOMS is a predictive model that gives the timing for a task as the sum of the timings for the individual operators used in the chosen method.
- GOMS has been shown to be beneficial for assessing the relative merits of various design prototypes
  - We have two designs, which one do we choose? A criterion could be according to GOMS (cognition, time)
- However, complex analysis can be very tedious, and tools are needed to automatize the process (e.g. CogTool)

PREDICTIVE MODELS - GOMS

GOMS (goals, operators, methods and selection rules)

- Example
  - Goal: Delete a piece of text from a Word document.

PREDICTIVE MODELS - GOMS

GOMS (goals, operators, methods and selection rules)

- Example
  - Operators
    - Left click
    - Move the mouse pointer
    - Select option from the menu
    - Press delete key

PREDICTIVE MODELS - GOMS

GOMS (goals, operators, methods and selection rules)

- Example
  - Method using the option from a menu
    - Select text to be deleted
    - The name of the option will be delete or erase
    - The option should be in Edit
    - Execute the option
    - Delete the text
  - Method using delete key
    - Mouse pointer before or after the text to delete
    - Remember which key is the delete key
    - Press key for deleting every letter
    - Delete the text

PREDICTIVE MODELS - GOMS

GOMS (goals, operators, methods and selection rules)

- Example
  - Selection rules
    - A sentence: mouse + menu
    - A word: delete key

PREDICTIVE MODELS - KLM

KLM (keystroke level model) evaluates the performance of a system.

Assigns a specific time to each action, except for the mouse (Pointing) that depends on the distance between the starting point and arrival.

- K = pressing a key or a button.
- P = pointing with the mouse to a target on the display.
- H = moving hands to the home position on the keyboard or mouse.
- M = a heuristic to be mentally prepared for a task.

Example: Searching a Microsoft Word document for all occurrences of a four-letter word, and replacing it with another four-letter word. Average typist = 55 wpm

PREDICTIVE MODELS - KLM

Example (Card and Moran (1980))

Description	Operation	Time (sec)
Reach for mouse	H[mouse]	0.40
Move pointer to "Replace" button	P[menu item]	1.10
Click on "Replace" command	K[mouse]	0.20
Home on keyboard	H[keyboard]	0.40
Specify word to be replaced	M4K[word]	2.15
Reach for mouse	H[mouse]	0.40
Point to correct field	P[field]	1.10
Click on field	K[mouse]	0.20
Home on keyboard	H[keyboard]	0.40
Type new word	M4K[word]	2.15
Reach for mouse	H[mouse]	0.40
Move pointer on Replace-all	P[replace-all]	1.10
Click on field	K[mouse]	0.20

PREDICTIVE MODELS - KLM

For a typical system there are 6 components:

- K (0.2 s) – press a key or mouse button
- P (1.1 s) – point with mouse
- H (0.4 s) – home on keyboard, mouse or other device
- M (1.35 s) – mentally prepare
- R (t) – system response time (needs to be measured)
- B (0.1s) – press or release mouse button
- BB (0.2s) – click mouse button

PREDICTIVE MODELS - KLM

Rules (simplified):

- M - when starting a task o when changing tasks
- H – when changing device (e.g. From mouse to keyboard)

- Example: MHPKPK
- Mental preparation, followed by a home (movement from keyboard to mouse), followed by two pairs of points and clicks on the mouse.

PREDICTIVE MODELS - KLM

Exercise – use a KLM method to predict the time that it will take to delete a file in a GUI environment in two different ways:

- Selecting and dragging the file con to the “recycle bin” icon.
- Selecting the file icon and using the “delete” option in the menu accessed by clicking the mouse right button.

To calculate the time, you can use the calculator in this web:

http://courses.csail.mit.edu/6.831/2009/handouts/ac18-predictive-evaluation/klm.shtml

PREDICTIVE MODELS

Fitt's law – the required time to reach a goal is proportional to the distance and the size of the goal.

- The main options must have a bigger size or be more visible than the secondary options.
- Location is also important: the four borders of a window are more easily accesible by users:
  - In Windows and Mac environments, the main options are usually placed in the menu bars at the sides of the screen.

Fitt's law

The index of difficulty (ID) is defined as the difficulty of a task in terms of distance and width. It is calculated as:

$$ID = \log_2 (D / W + 1), \text{ where}$$

- D is the distance to the center of the objective
- W is the width of the objective, measured on the movement axis. W also represents the tolerance to errors allowed at the final position, given that the final point of movement is at +/- W/2 from the center of the objective.

The mean time (MT) needed to reach an element of the interface is calculated as:

$$MT = a + b * ID, \text{ where}$$

- a represents the start/stop time in seconds for a given device
- b measures the inherent velocity of the device

PREDICTIVE MODELS

Hick's law – the required time to take a decision increases as the number of alternative options increases.

In mathematical terms, the time needed to choose an option among *n* alternative options is calculated as:

$$T = a + b \log_2(n + 1), \text{ where}$$

- a and b are coefficients determined experimentally.

This equation takes into account that all the options in the menu have the same probability to be chosen.

Hick's law

Some implications of Hick's law for web design

- Use saturated colors to draw attention to something.
- Use typography and neutral and balanced sizes if you want to choose between several options of the same level.
- Use movement and interaction to stand out powerfully.
- Use the balance of spaces. Do not place two elements of equal importance in places at a great distance from each other.
- Good semantics are essential when selecting words.

http://www.piensaenweb.com/blog/principios-del-diseno-web-iii-ley-de-hick---hyman_detalle_78.html

INTERACTION DESIGN

The process of interaction design: Life-cycles

What are the requirements?

Types of requirements

Data gathering

Interpretation and data analysis

Conceptual design

CONCEPTUAL DESIGN

- It aims to turn requirements and user needs into a conceptual model.
- A conceptual model is a high-level description of the proposed system, addressing what the system should do and how it should look like.
- Conceptual model – User interface – Mental model
  - Designers make a conceptual model and users make a mental model
- It is a phase highly linked to the gathering of information, but one should differentiate between what should be done and the solution adopted.

CONCEPTUAL DESIGN

Some guidelines:

- Keep an open mind, without forgetting the users and the context wherein the interaction takes place.
  - Strategy: think of them while building the conceptual design
- Aim for discussing your ideas with the users.
  - Our contact with the users do (should) not end after having gathered the requirements
- Design low-level prototypes to gather information quickly and engage the users in the discussion.
  - To have a more focused discussion
- Iterate, iterate, and iterate.

CONCEPTUAL DESIGN - Strategy

A three-pronged strategy:

- Interaction style & mode
- Metaphor
- Interaction paradigm

STRATEGY

Interaction style

- The type of interface and the interaction it implies, for example, command line, GUI, or speech

Interaction mode

- Based on the activities that the user will perform with the product
  - Instructions
  - Conversations (e.g. command language, form filling…)
  - Manipulation and navegation
  - Exploration (e.g. command language, menu selection…)
- Based on objects
  - Analogies

STRATEGY

Interaction mode

- What kind of activities does a game have?

- What kind of activities does a graphic package have?

STRATEGY

Interaction mode

- What kind of activities does a game have?
  - Manipulation and navigation
- What kind of activities does a graphic package have?
  - Instructions and conversation

STRATEGY

Other alternatives

- Process-oriented models: those applications where there is not a clear activity (e.g. Call center)

- Product-oriented models: those applications where products can be individually identified (e.g. Word document)

STRATEGY

Interaction mode

- In a system like Google Calendar, how can we carry out the task of making an appointment amongst various users considering their restrictions (time, relevance…)?
  - Conversation (which date? – I select the date)
    - Forms?
    - Command language?
    - Menu?
    - Speech?

METAPHOR

Metaphors are pervasive in everyday life, not just in language but in thought and action

ARGUMENT IS WAR

- Your claims are indefensible
- I demolished his argument
- You disagree? Okay, shoot!

Though there is no physical battle, there is a verbal battle, and the structure of an argument – attack, defense- reflects this

Lakoff, George, 1998 (2003). *Metaphors we live by,* London: The University of Chicago Press

METAPHOR

The essence of metaphor is understanding and experiencing one thing in terms of another

- ARGUMENT is structured in terms of WAR

Another type of metaphors are orientational

- A whole system of concepts is organised with respect to another concept
  - HAPPY IS UP; SAID IS DOWN
    - That boosted my spirits
    - I'm low these days
  - …

Lakoff, George, 1998 (2003). *Metaphors we live by,* London: The University of Chicago Press

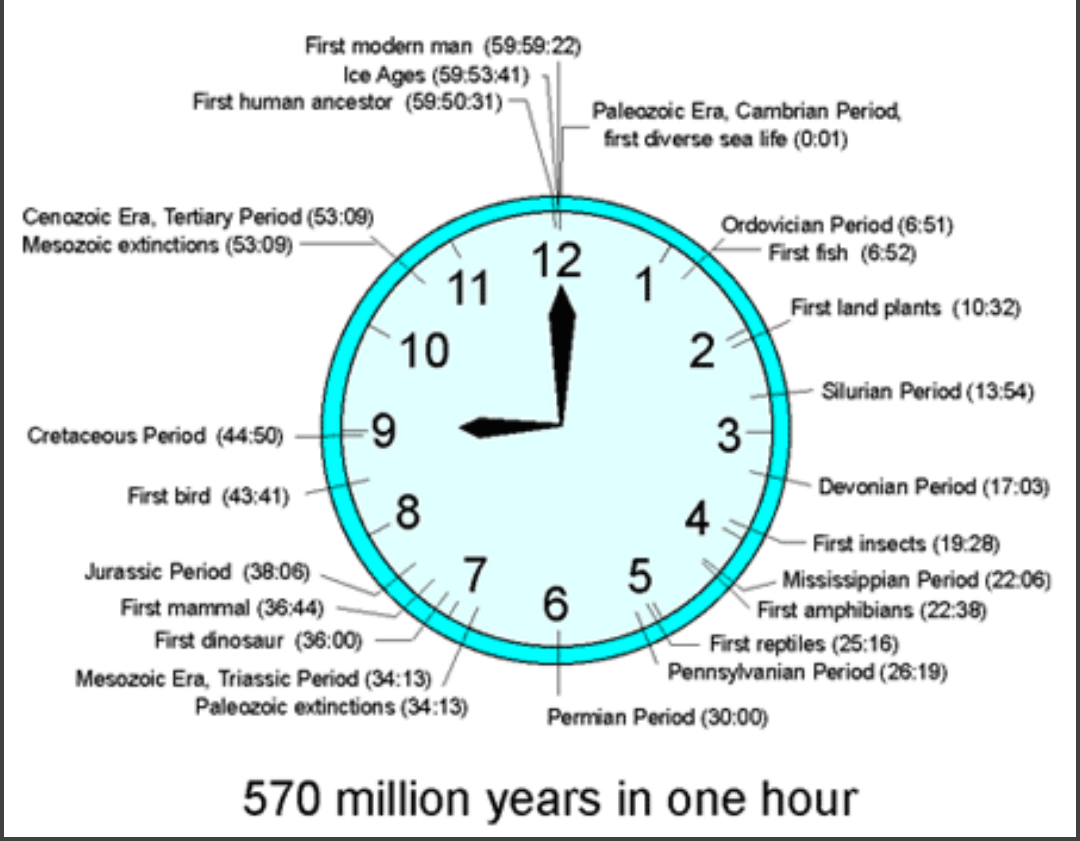
METAPHOR

Metaphors aim to:

- Combine familiar knowledge with new one, with the aim of helping the user to understand how to interact with the system
  - Desktop metaphor: we place documents in the desktop, we have folders on the desktop
- Balance the mixture of known and unknown elements

METAPHOR

Metaphor



METAPHOR

How to come up with a good metaphor?

- To understand what the system will do
- To identify what areas will be difficult to understand or give rise to problems
- To generate metaphors

Should answer questions such as:

- What is the structure of the metaphor? (folders, files…)
- Is it worth designing a metaphor? (can we do it in another way?)
- Is it possible to create the metaphor? (depending on the concept and amount of information)
- Will users understand the metaphor?
- Is it extensible/flexible? (desktop metaphors work in different operating systems…)

METAPHOR

How can we answer these questions with respect to the Google Calendar exercise?

- Structure → what structure does the metaphor provide?
- Relevance → how relevant is the metaphor?
- Representation → is it easy to represent the metaphor?
- Comprehension → will users understand the metaphor?
- Extensible → is the metaphor extensible?

METAPHOR

Another example: https://britishmuseum.withgoogle.com/

- Structure → what structure does the metaphor provide?
- Relevance → how relevant is the metaphor?
- Representation → is it easy to represent the metaphor?
- Comprehension → will users understand the metaphor?
- Extensible → is the metaphor extensible?

INTERACTION PARADIGM

Interaction paradigm

- A model or pattern of HCI that encompasses all aspects of interaction, including physical, virtual, perceptual, and cognitive.

- Following up on our calendar:
  - Ubiquitous/Pervasive computing
    - Computers considered a part of our lives.
  - Wearable computing
    - http://www.interaction-design.org/encyclopedia/wearable_computing.html

CONCEPTUAL DESIGN

We have developed a vision of the system

- What to do
- How to do it

Discuss / evaluate the conceptual design with your users

CONCEPTUAL DESIGN

More aspects to consider

- Technologies to be used: Multimedia, Virtual reality (VR), Augmented reality (AR)…
  - AR = Integration between real and virtual objects in a way that augments the user's perception and experience. The user must maintain a sense of presence in the real world

  http://www.youtube.com/watch?v=JSnB06um5r4
- User interfaces: Touch screen, voice recognition, …
- What concepts need to be communicated, how they relate, structure and are presented
  - What functions will the product do?
  - How are the functions related?
  - What information will be available?

CONCEPTUAL DESIGN

Example: In a course exercise delivery system, define:

- What functions will the product perform?
- How are the functions related?
- What information has to be available?

CONCEPTUAL DESIGN

Example: In a course exercise delivery system, define:

- What functions will the product perform?

	Functions
Administrator	Management of courses, degrees, subjects, ...
Teacher	Management of course exercises, ...
Student	Create group, introduce exercise, ..

CONCEPTUAL DESIGN

Example: In course exercise delivery system, define:

- How are the functions related?

Before doing…	Must…
Introduce course	Introduce course
Delete group	Be introduced
Introduce number of members	Select group

CONCEPTUAL DESIGN

Example: In a course exercise delivery system, define:

- What information has to be available?

To	Must know
Introduce course	Name, degree, coordinator name, course, ...
Delete group	ID or name of student
Introduce number of members	Number of group members for exercise