

Introduction à la Conception Centrée Utilisateur

IHM/CIE (NI226/NI205)

James Eagan
james.eagan@telecom-paristech.fr

JAMES EAGAN

MAÎTRE DE CONFÉRENCES EN INTERACTION HOMME-MACHINE

- Maître de conférences en IHM à Télécom ParisTech
- Ph. D., M. S. en Informatique, Georgia Institute of Technology
- B. A. en Mathématiques/Informatique, Lawrence University, Appleton, WI
- Recherche en Visualisation d'Information, Interaction Multi-surface



QUENTIN ROY

THÉSARD EN INTERACTION HOMME-MACHINE

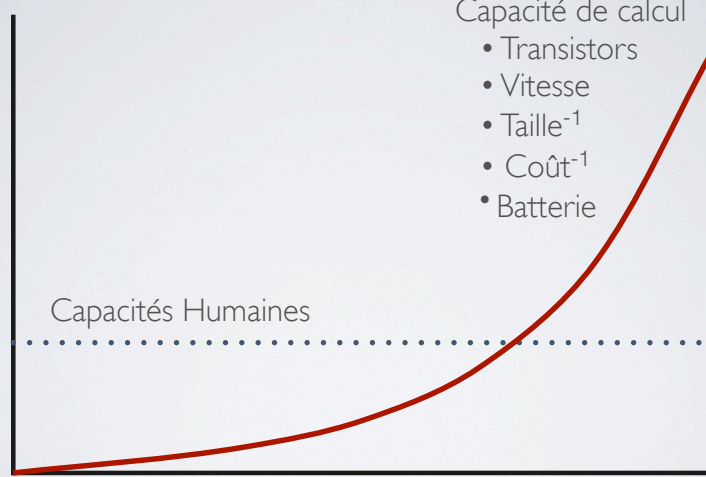
- Thésard en IHM à Télécom ParisTech
- Recherche en Interaction Gestuelle pour Surface Tactile



- 1999 : « La fonctionnalité la plus utilisée était ... recherche. Les personnes n'arrivaient pas à naviguer le site. »
 - « La deuxième fonctionnalité était le bouton 'help', car le moteur recherche était si inefficace. »
- Après re-conception du site centrée utilisateur :
 - Utilisation du bouton « help » a baissé 84 %
 - Ventes ont augmenté 400 %

[New York Times, 30 août 1999]

Loi de Moore



Capacité de calcul

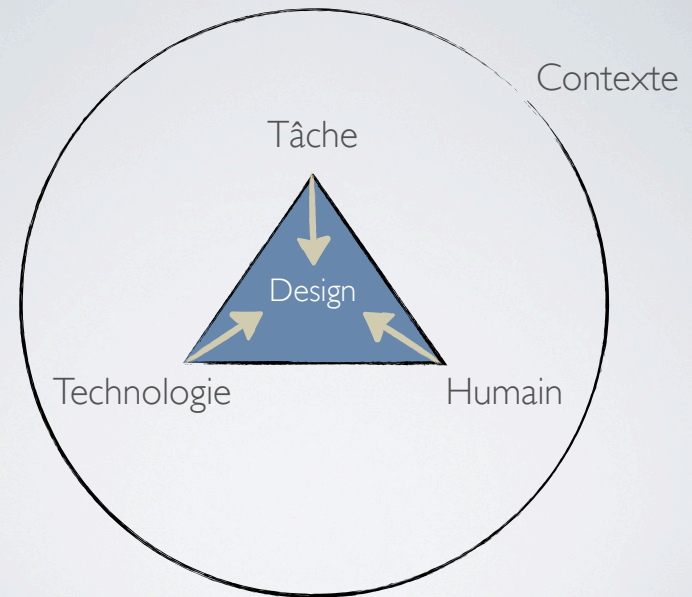
- Transistors
- Vitesse
- Taille⁻¹
- Coût⁻¹
- Batterie

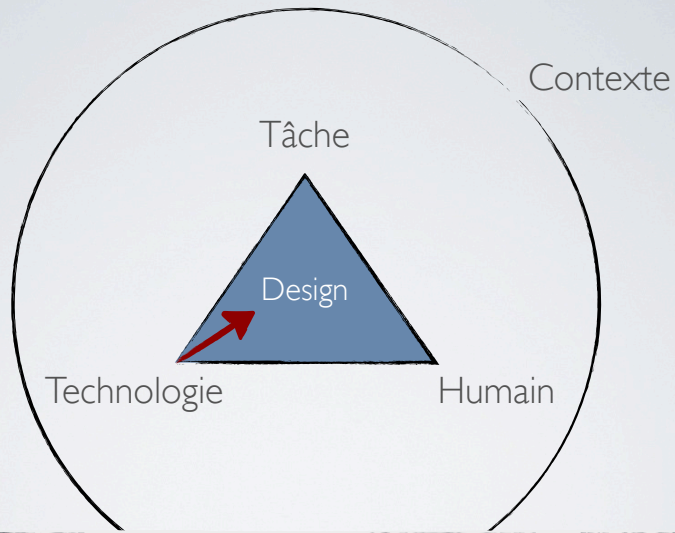
Capacités Humaines

INTERACTION HOMME-MACHINE

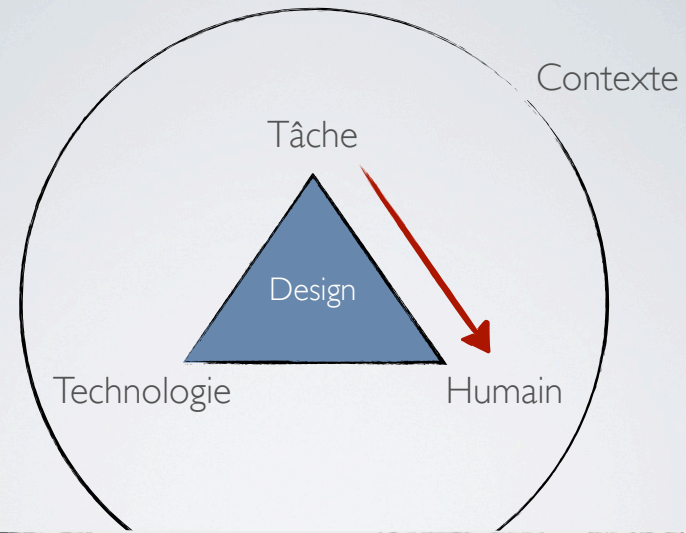
HUMAN-COMPUTER INTERACTION

- Human
 - L'utilisateur final du système
 - Autres personnes dans l'organisme
- Computer
 - La machine qui tourne le logiciel
 - Y en a souvent plusieurs
- Interaction
 - L'utilisateur exprime ce qu'il veut
 - L'ordinateur communique les résultats





Dans un smartphone avec écran tactile, on peut supposer une interaction multi-touch



Les connaissances d'une personne changent lorsqu'elle agit... elle apprend.

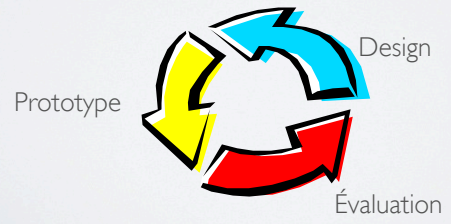
Interaction Homme-Machine

Un métier qui se concerne :

du *design*,
de l'*implementation*, et
de l'*évaluation*



de systèmes informatiques à usage par un être-humain



Conception de Design & Ingénierie de l'Utilisabilité

Buts :

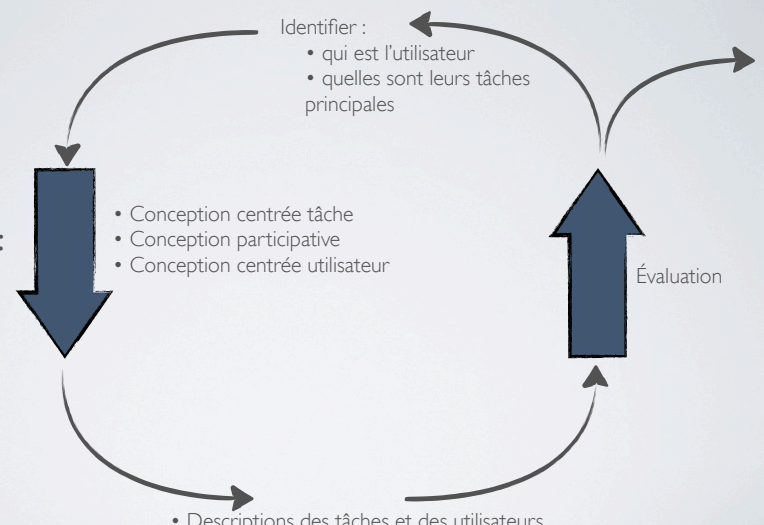
- Identifier :
- qui est l'utilisateur
 - quelles sont leurs tâches principales

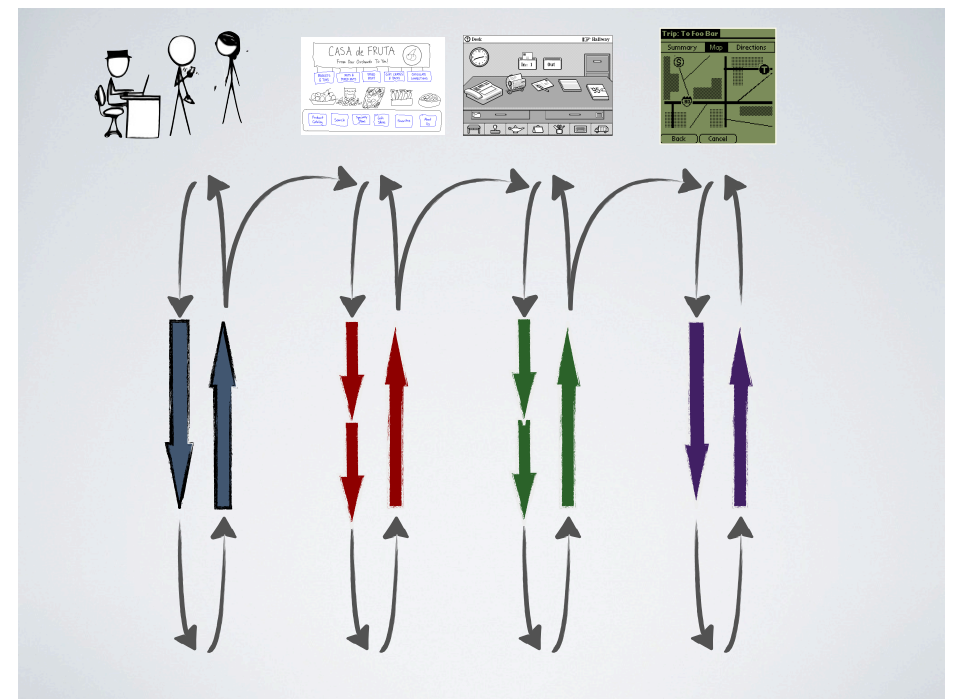
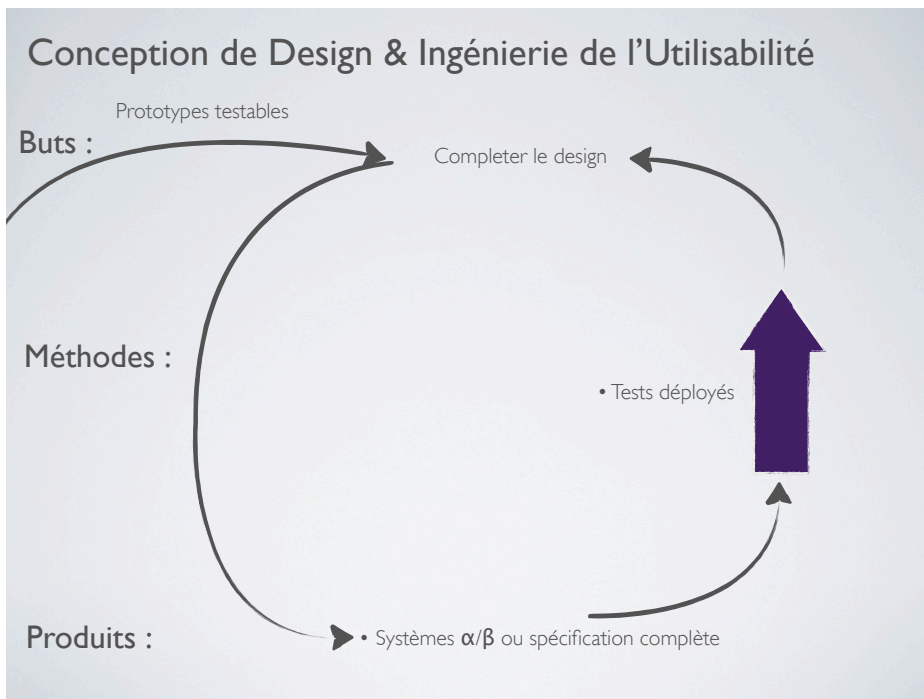
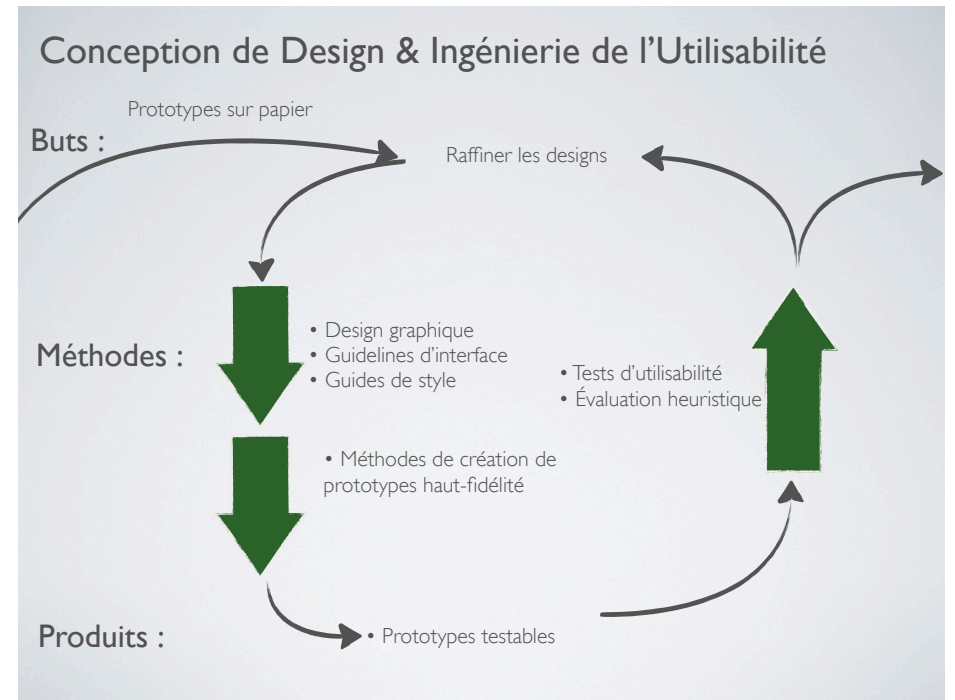
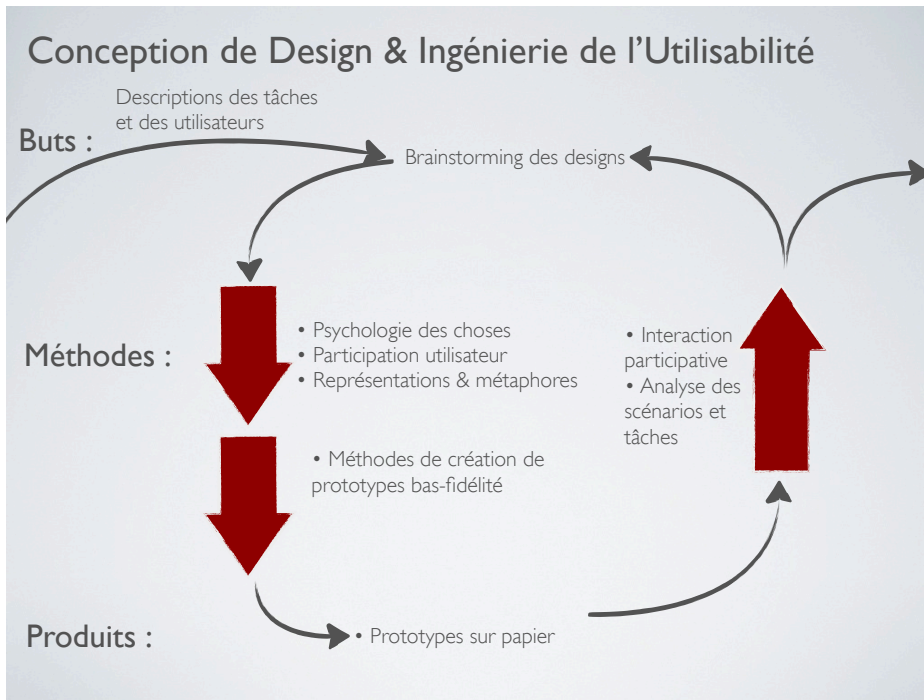
Méthodes :

- Conception centrée tâche
- Conception participative
- Conception centrée utilisateur

Produits :

- Descriptions des tâches et des utilisateurs





POURQUOI SUIVRE UNE MÉTHODE DE DESIGN ?

- 63% de gros projets expose leur budget
- 4 raisons principales :
 - Clients ont demandé des améliorations au design
 - Tâches/besoins loupées
 - Utilisateurs ne comprenaient pas leurs propres besoins
 - Communication entre développeur et utilisateur insuffisant

L'UTILISABILITÉ = GENIE LOGICIEL !

- ~Deux fois plus chère après déploiement
- Trop facile à commencer un design détaillé qui ... :
 - ... est basé sur des fausses suppositions des besoins
 - ... n'est pas suffisamment flexible
 - ... n'est pas facilement à utiliser
 - ... n'a jamais été testé

PRINCIPES DE BASES POUR LA CONCEPTION D'INTERFACES

Overview

- Introduction au cours et à l'IHM

Comprendre l'utilisateur et ses besoins

- Conception centrée tâche
- Développer des tâches indicatives
- Personae et scénarios utilisateurs

Design avec l'utilisateur

- Conception centrée utilisateur
- Conception participative
- Evaluer l'interface avec l'utilisateur



PRINCIPES DE BASES POUR LA CONCEPTION D'INTERFACES

Design d'interfaces visuelles

- Psychologie des choses de la vie quotidienne
- Au-delà du design sur l'écran
 - Représentations et métaphores
- Design graphique
 - Comment mettre en place les éléments graphiques sur l'écran

Principes de design

- Guidelines, heuristiques, etc.
- Comment découvrir des problèmes d'utilisabilité avant développement



OBJECTIFS DU COURS

Buts de ce module :

- Savoir comment récolter des besoins utilisateur et faire une analyse de tâches
- Avoir de l'expérience avec la conception centrée utilisateur
- Connaître plusieurs méthodes d'évaluation
 - ... quand les utiliser
 - ... pour quels buts

DÉROULEMENT DU MODULE

Travaux dirigés :

- 2 heures de cours suivies par 2 heures de TD

Devoirs à maison :

- À faire seule, à rendre avant la séance suivante

Projet :

- À faire par groupes de 5
- Une partie conception, une partie réalisation
- 50 % de la note finale
- Détails à suivre...

LA THÉORIE DE GRANDE UNIFICATION

- Qui est {l'utilisateur, le client, le sujet, ...} ?
 - Astuce : Il y en a probablement plusieurs
- Qu'est-ce qu'il ou elle essaye de faire ?
- Comment peut-on l'aider à le faire ? (Et gagner quelques €/\$/¥ en le faisant.)
- L'interface, réussit-elle ces buts ?

KNOW THY USER

- You are not your user
- Who are your stakeholders?
 - Travel system: employee, manager; auditor
- What is the user's goal?
 - How is success defined?
 - What are the constraints? Real-world, technical, political?
- User characteristics

REAL-WORLD CONSTRAINTS

- Time to market
- Cost/effort to design & implement
- Size/footprint/weight/price/power
- Computer power/memory
- Consistency with product line/brand image
- Backward compatibility
- Differentiation from competitive products

HOW TO UNDERSTAND THE USER

- Gather data
 - Interviews, observation, surveys & questionnaires, documentation, immersion
- Organize data
 - Notes, cards, brainstorming, computer tools
- Represent data
 - Lists, outlines, matrices
 - Narratives, Scenarios
 - Hierarchies, Networks, Flow Charts

WHAT TO GATHER

- Three key components in how people work
 - Activities
 - Artifacts
 - Relations
- Not just computer system oriented!
- The context matters!
 - Office: papers, whiteboards, ...
 - Phone calls: address book, note pad, dialer, ...

FOCUS ON *OBSERVABLE BEHAVIORS*

- What are the practices, methods, steps, objects, ..., used?
- Learn *what* users do, *why* they do it, *how*, they do it, *when* they do it, with what *tools* or *people* they do it
 - Your new system may change some of this, especially *how*
 - Understanding the *how* and the *why* is what leads to deeper knowledge and insights

DATA GATHERING

- Tasks & Subtasks
 - Physical
 - Cognitive
 - Communication
- Conditions under which these are done
- Results/outcomes of tasks

TASK REQUIREMENTS

- Requirements to perform task
 - Information
 - Communication with others
 - Equipment

Must include
Should include
Could include
Exclude

SOME DATA GATHERING METHODS

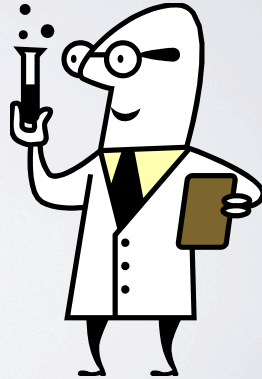
- Observation & Think-aloud
- Cooperative Evaluation
- Interviews
- Questionnaires & Surveys
- Focus Groups
- Study Documentation
- Competitive Product Analysis
- Ethnography

INTERPRETIVE ANALYSIS

- Controlled Experiments: formal & objective
- Interpretive Analysis: more subjective
 - Concerned with humans, so no objective reality
 - Sociological approach

CONTROLLED EXPERIMENTS

- Great for objective performance
- But:
 - Lab is not the real world
 - Can't control all the variables
 - Context matters
 - Tasks are short, artificial



INTERPRETIVE ANALYSIS METHODS

- Field studies, contextual inquiry, ethnography
 - Inspired by anthropology (€¥\$)
- Interviews
- Philosophy:
 - Formal environment of controlled study is artificial, so
 - Get into user's environment
 - Interpretation is primary over data

OBJECTIVES

- Understand the user
 - What are his or her goals & values?
 - Individual's or group's interactions within a culture
 - Make *tacit* domain knowledge explicit
 - Be *unbiased*
 - For UI designers: improve system by finding existing problems

TECHNIQUES

- In-person observation
- Audio/video recording
- Interviews

- "Wallow in the data"

OBSERVATION IS KEY

- Carefully observe everything about the users and their environment
- Think of describing it to someone who has never seen this activity before
- What users say is important, but also non-verbal details

OBSERVATIONS

- Things of interest to the evaluator
 - Structure & language used in work (domain vocabulary)
 - Individual & group actions
 - Work culture
 - Explicit & implicit aspects of work
- Example: Office environment
 - Business practices, rooms, artifacts, work standards, relationships between workers, managers, ...

INTERVIEWS

- Have a question plan, but keep interview open
- Be specific
- Create interpretations together with users
 - Be sure to use *their terminology*
- At the end, ask if there's anything else you should have asked
- Record interviews

STEPS

- I. Preparation
 - Understand the organizational context
 - Familiarize yourself with system and its history
 - Set initial goals and prepare questions
 - Gain access and permission to observe & interview

DURING INTERVIEWS

- Establish rapport with users
- Observe/interview users in workplace and collect all different forms of data
- Follow any leads that emerge from visits
- Record the visits

ANALYSIS

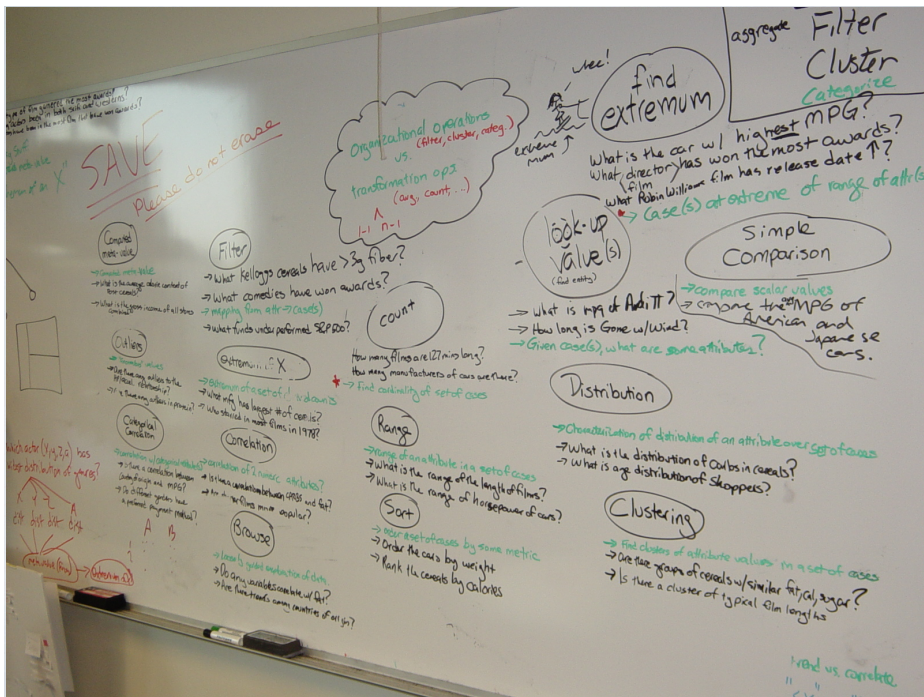
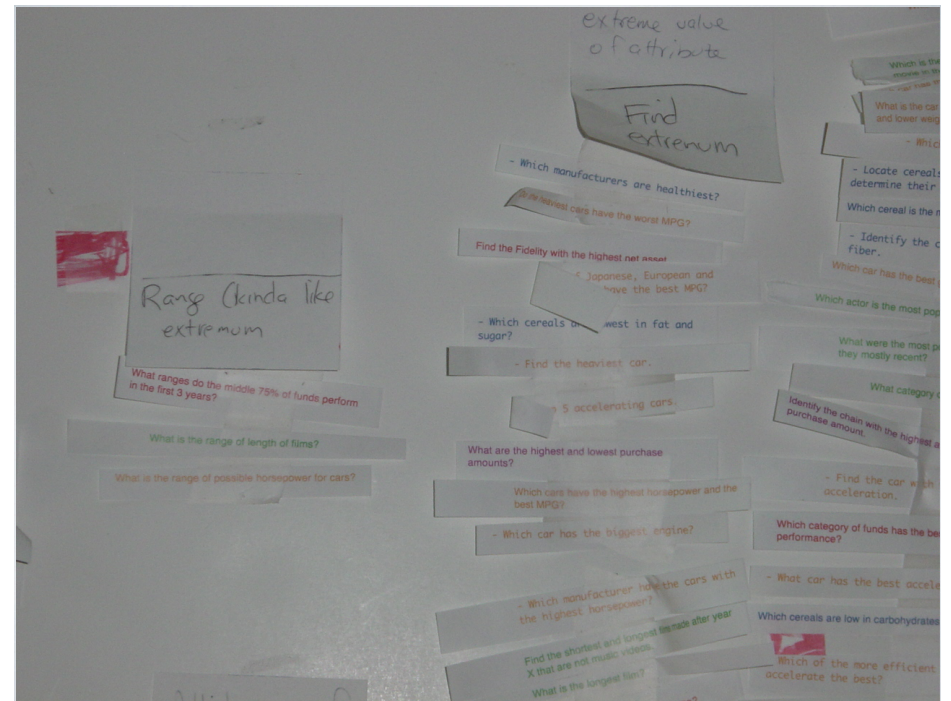
- Compile the data in numerical, textual, and multimedia databases
- Quantify data and compile statistics
- Reduce and interpret data
- Refine goals and process used

REPORTING

- Consider different audiences and goals
- Prepare a report and present findings

AFFINITY DIAGRAM

- Useful technique for qualitative data analysis
- Write each observation/quote on a slip of paper
- Put it on a board/wall
- Coalesce items that have affinity
- Give names/colors to groups
- Continue making subgroups
- May yield a hierarchy of groups



WHY IS THIS USEFUL?

- Can help gain a rich and true assessment of user needs
- Helps to define requirements
- Uncovers true nature of user's needs
 - Discover things that are outside job description, documentation
- Allows you to put yourself in the role of an end-user
- Open-ended and unbiased nature promotes discovery

TYPES OF FINDINGS

- Qualitative
 - Observe trends, habits, patterns, ...
- Quantitative
 - How often was something done, what percent of the time did something occur, how many errors, ...

DRAWBACKS

- Takes a lot of time
- Scale : small numbers
- Qualitative results are subjective and difficult to generalize
- Acquired skill
 - Identifying and extracting meaningful and “interesting” things is challenging

ALTERNATIVE: OBSERVATIONS

- Observe user performing activity of interest to you
- Record audio/video/screen (with permission)
-

INTERVIEWS

- Structured — “Just the facts”
 - Efficient
 - Training: interview process
- Unstructured — A conversation
 - Inefficient
 - Training: process + domain knowledge

SEMI-STRUCTURED INTERVIEWS

- Start with focused questions, move to open-ended discussion
 - Good balance, often appropriate
 - Training: process + domain knowledge

SEMI-STRUCTURED INTERVIEW QUESTIONS

- Pre-determine data of interest — know why you are asking questions, don't waste time
- Plan for effective question types
 - How do you perform task x?
 - Why do you perform task x?
 - Under what conditions do you perform task x?
 - What do you do before you perform...?
 - What information do you need to...?
 - Whom do you need to communicate with ...?
 - What do you use to...?
 - What happens after you...?
 - What is the result or consequence of...?
 - What is the result or consequence of NOT...?
- See Gordon & Gill, 1992; Graesser, Lang, & Eloffson, 1987

TYPICAL OPEN-ENDED QUESTIONS

- Why do you do this (whatever the task is you are studying)?
- How do you do this?
 - Gets at task-subtask structure
 - Then ask about each subtask
- Why do you do it this way rather than some other way?
 - Attempts to get user to explain method and rationale so you can assess importance of the particular way of doing task (onion)

MORE OPEN-ENDED QUESTIONS

- What has to be done before you can do this?
 - To get at sequencing issues
- Please show me the results of doing this
- Do errors ever occur when doing this?
- How do you discover the errors, and how do you correct them? (Adapted from Nielsen et al., CHI '86).
- Encourage digressions; ask for elaborations
- What else should I have asked you?

QUESTIONNAIRES

- General Criteria
 - Make questions clear & specific
 - Ask some closed questions with range of answers
 - Sometime also have a *neutral* or *other option*
 - Do test run with one or two people

LIKERT SCALE

Evaluation Questionnaire

Please complete the following questionnaire by indicating how strongly you agree or disagree with the following statements. Your responses will be kept confidential and will be used only for improving the interface that you worked with in this experiment.

1. I felt that the computer agent's help was **worthwhile**.
1-----2-----3-----4-----5-----6-----7
Strongly Disagree Strongly Agree
2. I found the computer agent to be **intrusive**.
1-----2-----3-----4-----5-----6-----7
Strongly Disagree Strongly Agree
3. I found the computer agent's **help** to be **distracting**.
1-----2-----3-----4-----5-----6-----7
Strongly Disagree Strongly Agree

- Seven-point Likert scale (use odd #)
- Could also just use words (e.g., strongly agree, agree, neutral, disagree, strongly disagree)

OTHER TYPICAL QUESTIONS

- Rank the importance of each of these tasks
- List the four most important tasks that you perform (this is an open question)
- List the pieces of information you need to have before making a decision about X, in order of importance
- Are there any other points you would like to make? (open-ended opinion question; good way to end)
- Same questions can be used in interview and in questionnaire; difference is in follow-up opportunity

FOCUS GROUPS

- Group of individuals — 3 to 10
 - Use several different groups with different roles or perspectives
 - And to separate the dominant personalities from the others
 - Want to avoid few people dominating discussion
- Use structured set of questions
 - More specific at beginning, more open as progresses
 - Allow digressions before coming back on track

STUDY DOCUMENTATION

- Describes how things should be done rather than how they are done
- Try to understand these discrepancies

COMPETITIVE ANALYSIS

- Look at competing products
- Look for both good and bad ideas
 - Functionality
 - UI Style
- Do user task performance metrics to establish bounds for your system

WHICH METHOD TO USE?

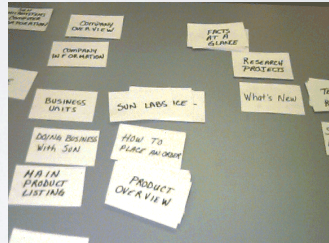
- Depends on your own resources ...
 - Current knowledge of tasks & users
 - Context
 - *E.g.*, can't use think-aloud if tasks involve two people working together

ORGANIZING OBSERVATIONS

- Organizing the observations serves two purposes
 - Understand the data
 - Helps present the data

TOOLS FOR SENSEMAKING

- Card sorting – to create Affinity Diagrams
- Also useful for web site organization
- Do it with multiple users
- Flow charts
- Task analysis diagrams



NOW WHAT?

NOW WHAT?

- You have piles of notes, hours of video, surveys up to here...
- How can you digest and represent the *data*, to turn it into *information*?

REPRESENTING DATA

- Essential use cases
- User characteristics + personae
- Task outlines
- Narratives
- Hierarchies & Network Diagrams
- Flow Charts

ESSENTIAL USE CASE (SCENARIO)

- Description of important or frequent user interactions
- Used to evaluate/walkthrough various design alternatives
- Three elements
 - Name
 - User intention
 - System responsibility
- *Do not make assumptions about the UI design*

EXAMPLE USE CASE

Arrange-Meeting

- User Intention
 - Arrange a meeting
 - Identify meeting attendees and constraints
 - Choose preferred date
- System Responsibility
 - Request attendees & constraints
 - Suggest potential dates
 - Book meeting

Get foreign currency	
User's purpose	System responsibility
Identify self.	Validate user's identity. Display currencies available.
Select currency required.	Display exchange rate.
Enter amount of foreign currency required.	Calculate exchange (e.g. £ to \$).
Confirm amount.	Request initiation of payment. Obtain authorisation for amount. Give money.
Take money and go.	

[From *User Interface Design and Evaluation*, The Open University]

USER CHARACTERISTICS & PERSONA

- Description of user and what he or she wishes to do
- Be specific/detailed, even give names and picture
- Three persona for ATM usage follow
 - Adapted from *User Interface Design & Evaluation*, The Open University
- Developed by Cooper (1999)

FELIX (TEENAGE ATM USER)

- Felix is 13 and gets pocket money each week. He spends it with his friends, so doesn't make regular deposits. He does receive gifts for his birthday, Christmas, etc. and saves that money for special purchases, such as a computer games console or trendy clothes. He has an ATM card allowing him to make withdrawals when needed for his purchases.



AGE: 28-55

“Most of my time is spent figuring out if they have enough funds to run the file.”

Alex has a ritual every morning—go through email and look for messages from finance about his regular clients he needs to process for that day. Then he signs into BMC, goes to the AJL, and starts working through his clients' files one-by-one. This is a process he repeats pretty much all day long.

As the day continues, more files from clients come in. In any given day, he might process 20-30 files. The system “works,” but there are definitely areas Alex would like to see improved. For example, each file is typically a funding order, or a new card order, but those details are never available on the AJL—he has to click a link every time to see them.

Fortunately, he knows what to expect from his regular clients, but that still leaves him clicking a “detail” link about 10-15 times a day. And the AJL never shows whether or not funds are available. Instead, he has to go to into his email, look for an email from Finance with the Client Invoice number, then go into Great Plains, which takes FOREVER to load, find the invoice number, locate the job number, then back into the AJL, find the job number and give it the green light. And for some reason, he can't process more than one file from the same client at a time. So, he has to keep checking back.

At any given time, he's bouncing between 6-8 different windows. The ideal system for Alex? Well, he could skip Great Plains—just have the system tell BMC, which jobs have funds and are ready to run. Also, it should just his clients and take him right to the AJL.



The diagram shows a vertical axis for 'Knowledge' and 'Activities and Interest' with various icons representing different tasks and systems. Below the axis is a 'Lifecycle' bar with 'Jan' and 'Jul' markers, and a 'Date' axis with 'Month' labels.

<p>Key Characteristics</p> <ul style="list-style-type: none"> Finding his files and processing them. Primarily works in Active Job List. Some work in Submitted Job List. Manages multiple clients at once. Serve as secondary contact with client. Determine if client has funds available to run their files. Learn their clients' schedules over time. Balances taking care of high-profile clients with lower-profile clients with immediate needs. 	<p>Goals</p> <ul style="list-style-type: none"> Be able to see just their client files. Quickly respond to client issues. Better integration with finance to show funds are available. Possibly auto process files with available funds. Spend less time doing grunt work, things that could be automated and auto-populated from the system. Balancing taking care of high profile clients with low profile clients that have an immediate need. <p>Questions</p> <ul style="list-style-type: none"> Who are my clients? Are there funds available? What files are in my queue? Can the file be processed now? 	<p>Influencers</p> <ul style="list-style-type: none"> Maintained. Availability of necessary information (e.g. financial, schedule). <p>Applications</p> <ul style="list-style-type: none"> BMC (AJL, SJL) Email Great Plains Report Generator CSA Mantis 	<p>Frustrations & Pain Points</p> <ul style="list-style-type: none"> Default view shows all files, when all he needs to his. Too much switching between BMC, Outlook, Great Plains, and Report Generator. No initial screen to show him what's in his queue for the day, or recent activity. Has to go through an extra screen before coming to his AJL. No file details available on screen, has to click details to see what kind of transaction it is (e.g. cardholder order, funding). Great Plains is painfully slow.
--	---	--	---

SANDRA (YOUNG ADULT)

- Sandra is 30, is married to Jason, has two children Todd(6) and Carly (18 months). They live in a subdivision that is about three miles from the town center, where the bank and stores are located. Jason uses the car for work, and works long hours, leaving at 6:45 am and returning at 8:00 pm. Sandra does not drive, so has to use public transportation. She tries to run errands and shop while Todd is in school, so she does only has to take Carly to town with her. She typically needs to make two trips to town each week to get everything done. She uses a stroller with Carly, and the bank is one flight up via escalator, so she prefers to use the ATM outside the first floor, even though there is no canopy to protect customers from bad weather.

GRANDPA MARVIN (OLDER ADULT)

- Marvin is 68 years old, and his social security is deposited into his bank account at the start of each month. He goes to the bank every week, withdrawing enough cash for the week - for miscellaneous expenditure. Regular bills are paid by check. He stands in line for a live teller, as he prefers the social interaction to using an ATM, even though his new artificial hip makes standing in line uncomfortable. He does not have an ATM card.

TASK OUTLINES

- Lists, outlines, matrices
 - Use expanding/collapsing outline tool
 - Add detail progressively
 - Know in advance how much detail is enough
 - Can add linked outlines for specific subtasks
- Good for sequential tasks, not so good for parallel

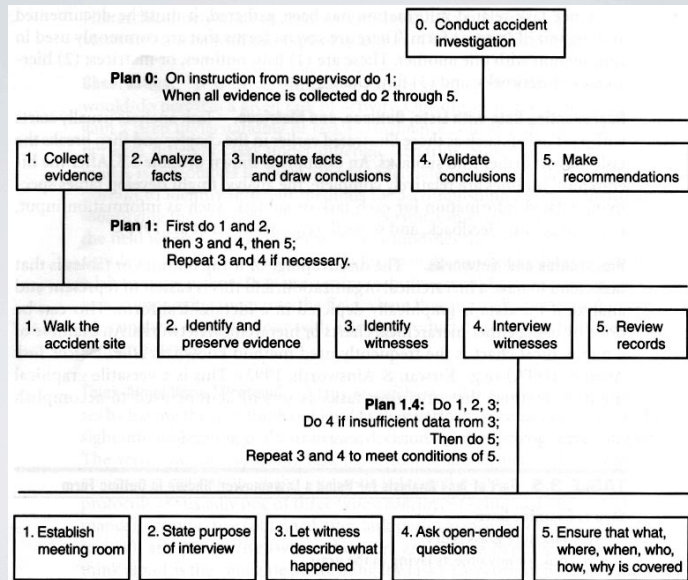
- Using a lawnmower to cut grass
 - Step 1. Examine lawn
 - Make sure grass is dry
 - Look for objects laying in the grass
 - Step 2. Inspect lawnmower
 - Check components for tightness
 - Check that grass bag handle is securely fastened to the grass bag support
 - Make sure grass bag connector is securely fastened to bag adaptor
 - Make sure that deck cover is in place
 - Check for any loose parts (such as oil caps)
 - Check to make sure blade is attached securely
 - Check engine oil level
 - Remove oil fill cap and dipstick
 - Wipe dipstick
 - Replace dipstick completely in lawnmower
 - Remove dipstick
 - Check that oil is past the level line on dipstick
 - ...

NARRATIVES

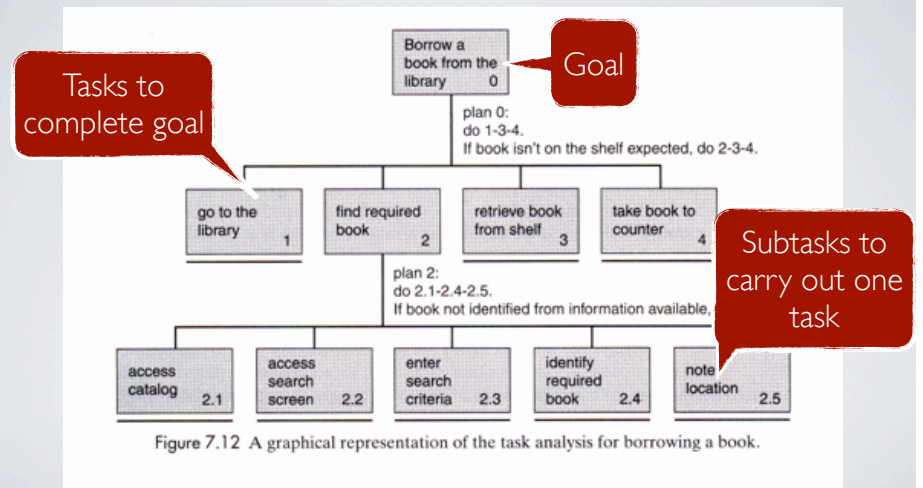
- Describe tasks in sentences
- Often expanded version of list or outline
- More effective for communicating general idea of task
- Not effective for
 - details
 - branching tasks
 - parallel tasks
- Great as introduction to diagrams or outlines

HIERARCHIES & NETWORKS

- Goals – what the user wants to achieve
- Tasks – do these to achieve the goals
 - Sequential dependencies
 - Create new document before entering text
 - Multiple occurrences of tasks
 - Subtasks – lower-level tasks
- The lowest-level subtasks get mapped onto one or several UI commands
- *i.e., move done by a copy followed by a paste*

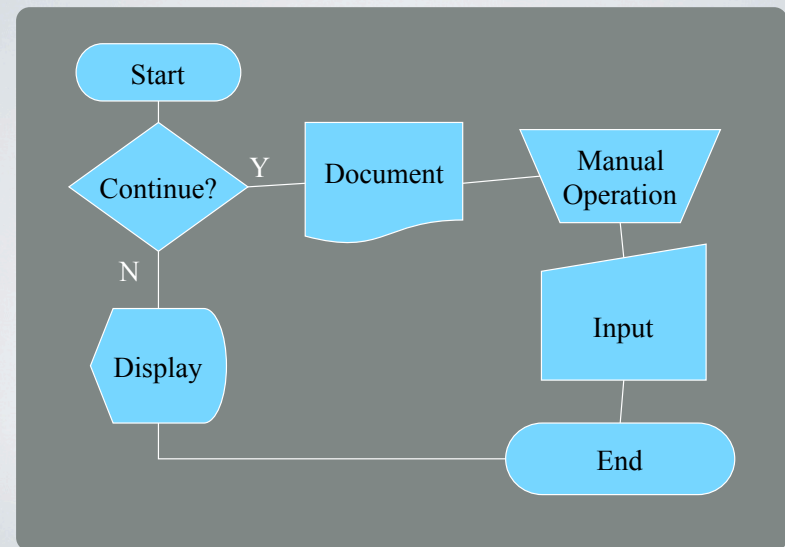


TASK MODEL — BORROW A BOOK



CAN BE MORE THAN ONE

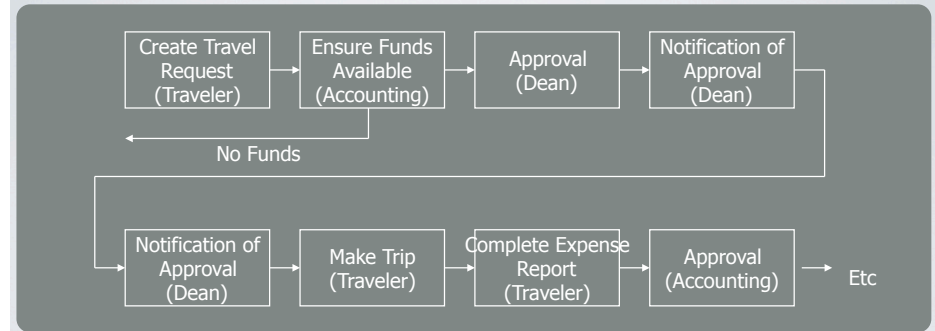
FLOWCHARTS



WORKFLOWS

- Documents going from one person/organization to another
- Multiple participants in an activity
- Web page sequencing
 - Browsing, purchasing, checkout

DOCUMENT FLOW EXAMPLE



MULTIPLE PARTICIPANTS

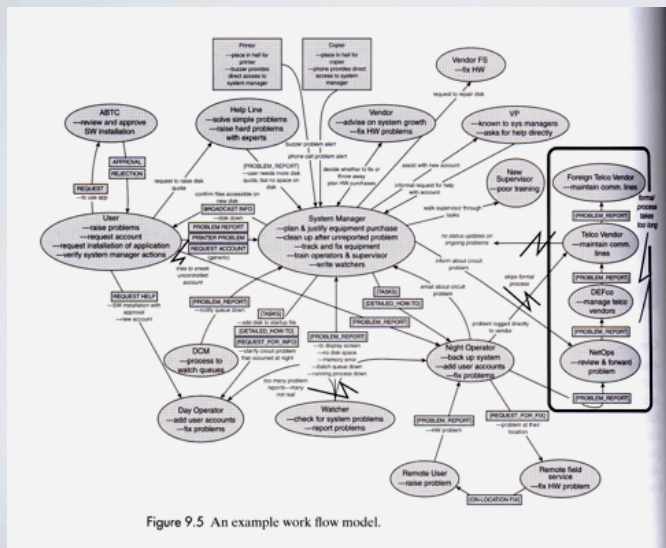


Figure 9.5 An example work flow model.

[From Interaction Design, Preece Rogers and Sharp]

SUMMARY OF TASK ANALYSIS

- Determine the data you need
- Gather it using various appropriate methods and techniques
- Represent the tasks and subtasks, plus other related information
- Use this data to improve design
- Note: be efficient!

USING WHAT YOU'VE LEARNED

- How do attributes of users & their tasks influence the design of user interfaces?
- Are there some design guidelines we can derive from different attributes?

USER PROFILES

- Attributes:
 - attitude, motivation, reading level, typing skill, education, system experience, task experience, computer literacy, frequency of use, training, color-blindness, handedness, gender, ...
- Novice, intermediate, expert
- Manager, employee, contractor, ...

MOTIVATION

- | <u>User</u> | | <u>Design goal</u> |
|-------------------------------------|---|---|
| • Low motivation, discretionary use | → | • Ease of learning |
| • Low motivation, mandatory use | → | • Control, power |
| • High motivation, due to fear | → | • Ease of learning, robustness, control |
| • High motivation, due to interest | → | • Power, ease of use |

KNOWLEDGE & EXPERIENCE

Experience

<u>task</u>	<u>system</u>	<u>Design goals:</u>
low	low	– Many syntactic & semantic prompts
high	high	– Efficient commands, concise syntax
low	high	– Semantic help facilities
high	low	– Lots of syntactic prompting

JOB & TASK IMPLICATIONS

- Frequency of use
 - High — Ease of use
 - Low — Ease of learning & remembering
- Task implications
 - High — Ease of use
 - Low — Ease of learning
- System use
 - Mandatory — Ease of use
 - Discretionary — Ease of learning



DEFINE TASKS

- Consider the *whole* system
- Determine *who* or *what* should perform each task and each step :
e.g. the system remembers the login, but the user remembers the password
- Determine criteria: efficiency, cognitive effort, time
 - Task x should take no more than y seconds
 - A new user should be able to create a new account in 5 minutes

PROJET

INTERFACE TACTILE POUR UN RESTAURANT

- Un restaurateur vous approche demandant de créer une interface tactile pour tablette (iPad, Galaxie) pour mettre à disposition des clients dans un restaurant pour prendre les commandes.
 - Vous allez faire un récolte de besoins et analyse de tâches
 - Créer plusieurs prototypes bas-fidélité
 - Construire un prototype haut-fidélité
 - L'évaluer