Évaluation d'interface, critères ergonomiques

Aurélien TABARD
aurelien.tabard@liris.cnrs.fr

Plan des prochaines séances

- **Introduction**
- **Approches d'évaluation**
- **Méthodes analytiques**

Plus tard dans le cours :

- Évaluation 2.0 : passer à l'échelle

Évaluation d'interface, critères ergonomiques

- **Introduction**
- Approches d'évaluation
- Méthodes analytiques

Plus tard dans le cours :

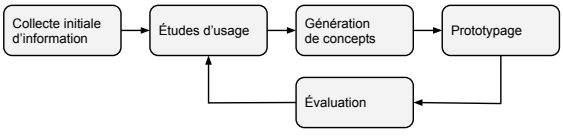
- Évaluation 2.0 : passer à l'échelle

Le rôle de l'évaluation

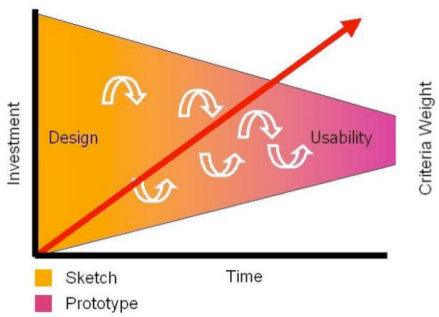
- Une partie du cycle itératif : *design-build-evaluate*
- Une comparaison entre ce qui est "construit" et ce qui était prévu
- Un endroit pour réfléchir sur cette différence et la prochaine phase de conception

From Sketching User Experiences by Bill Buxton

Quand évaluer



Le processus de conception



Être agile

Rater rapidement pour réussir plus tôt :

- Itérations sur des prototypes basse fidélité
- Design en parallèle : construire et tester plusieurs prototypes
- Explorer des alternatives

Augmenter progressivement la fidélité

Affronter la réalité, concevoir pour des cas d'utilisation pas des spécifications

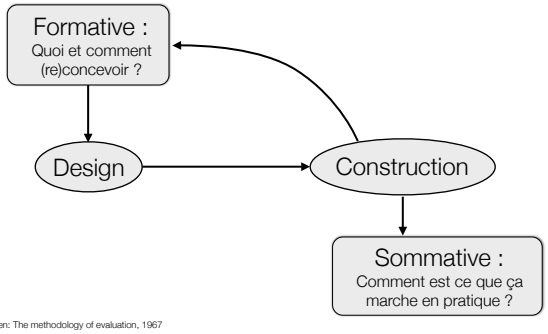
Évaluation d'interface, critères ergonomiques

- Introduction
- **Approches d'évaluation**
- Méthodes analytiques

Plus tard dans le cours :

- Évaluation 2.0 : passer à l'échelle

Évaluation Formative ou Sommative ?



M. Scriven: The methodology of evaluation, 1967

Évaluation Analytique vs. Empirique

"If you want to evaluate a tool, say an axe, you might study the design of the bit, the weight distribution, the steel alloy used, the grade of hickory in the handle, etc., or you may just study the kind and speed of the cuts it makes in the hands of a good axeman."

[Scriven, 1967]

Des méthodes complémentaires

L'évaluation empirique permet de comprendre les implications des propriétés de l'objet

- La hache va elle trancher tel buche ?

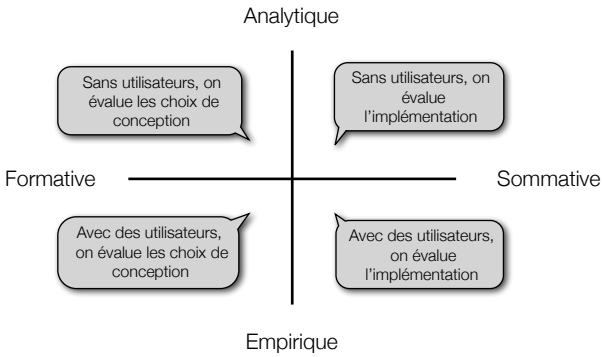
L'évaluation analytique identifie offre une grille critique sur les propriétés importantes

- Le manche de la hache est il compatible avec les gauchers ?

Dans les deux cas :

- Productions de faits qui doivent être interprétés

Des approches orthogonales



Une évaluation sans critère ne sert à rien !

If faut définir ce qu'on veut savoir avant d'évaluer !

Critères possible (parmi bien d'autres) :

- Test informel d'une idée contre une autre
- Analyse statistique de la performance moyenne
- Acceptation par un groupe d'utilisateur réaliste
- Vérification de critères heuristiques / ergonomiques
- Mesure de métriques d'utilisabilité lié au design

Évaluation d'interface, critères ergonomiques

- Introduction
- Approches d'évaluation
- **Méthodes analytiques**

Plus tard dans le cours :

- Évaluation 2.0 : passer à l'échelle

Les grands types d'évaluation analytique

Basée sur des modèles

- Évaluation selon des modèles d'interaction

Basée sur l'inspection

- Review d'experts / critique du design
- Cognitive walkthrough
- Évaluation heuristique

Inspections et critiques d'experts

- Tout au long du processus de développement
- Conduite par des développeurs et des experts (internes ou externes)
- Outil pour identifier des problèmes
- Peut aller d'une heure à une semaine de travail
- Préférer une approche structurée
  - Les reviewers doivent pouvoir communiquer sur tous les problèmes (sans fâcher l'équipe)
  - Les critiques ne doivent pas être aggressive envers les développeurs / designers
  - L'objectif principal est d'identifier les problèmes (pas leur source)
- Des solutions peuvent être suggérées a l'équipe

Méthodes d'inspection

Inspection des "Guidelines"

- Vérifier que l'interface respecte bien un ensemble de règles.

Inspection centrée cohérence

- Vérifier que l'interface est cohérente / consistante avec elle-même, avec les applications liées, avec l'OS
- Une vue générale peut aider, ex : impression des pages clés du site collées au mur.
- On peut forcer la consistance par les outils, ex : via les CSS pour les sites Web

Procédure d'inspection

- Trouver des experts/reviewers
- Définir un plan avec des limites temporelles
- Préparer le materiel pour les reviewers, y compris les critères d'intérêt
- Sur place ou sur un autre site
- Rédaction d'un rapport et définition des conséquences

+/- des évaluations par experts

- Results of informal reviews and inspections are often directly used to change the product
  - ... still state of the art in many companies!
  - The personal view of the CEO, or his partner ...
- Really helpful evaluation
  - Is explicit
  - Has clearly documented findings
  - Can increase the quality significantly
- Expert reviews and inspections are a starting point for change

Les critères ergonomiques (usability guidelines)

- Don Norman's principles:
  - visibility, affordances, natural mapping, and feedback
- Ben Shneiderman's 8 Golden Rules of UI design
- Bruce Tognazzini's 16 principles:
  - http://www.asktog.com/basics/firstPrinciples.html
- Christian Bastien's Ergonomic Criteria
- Jakob Nielsen's Heuristics

L'évaluation heuristique

Conçue comme une méthode d'évaluation "discount" basée sur l'inspection :

- Évaluation rapide, pas cher et facile d'interfaces
- http://www.useit.com/papers/heuristic/

Principes :

- Il y a une liste de propriétés désirable dans une interface : les "heuristiques"
- Ces heuristiques peuvent être vérifiées par des experts avec un résultat clair et précis

10 Usability Heuristics

http://www.useit.com/jakob/photos/

- Meet expectations
  1. Match the real world
  2. Consistency & standards
  3. Help & documentation
- User is boss
  4. User control & freedom
  5. Visibility of system status
  6. Flexibility & efficiency
- Errors
  7. Error prevention
  8. Recognition, not recall
  9. Error reporting, diagnosis, and recovery
- Keep it simple
  10. Aesthetic & minimalist design

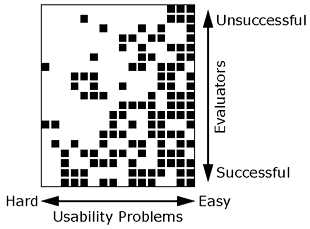
Procédure

- Un petit nombre d'évaluateurs examine l'interface en utilisant et juge de son adéquation aux critères d'utilisabilité (les "heuristiques").
- Soit par inspection, soit au travers d'un scénario
- Les problèmes sont listés et organisé par sévérité
- Les opinions des évaluateurs sont synthétisées dans un rapport.

Quantité d'évaluateurs

Chaque évaluateur ne trouve pas tous les problèmes

Les bons évaluateurs trouvent à la fois des problèmes faciles et d'autres plus difficiles

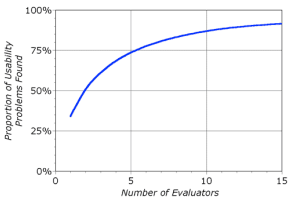


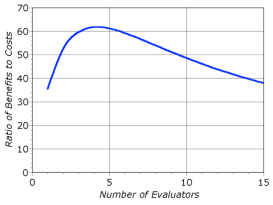
Quantité d'évaluateurs

Un évaluateur unique fournit de mauvais résultats

Trouve seulement 35% des problèmes d'utilisabilité

5 évaluateurs trouvent ~ 75% des problèmes d'utilisabilité



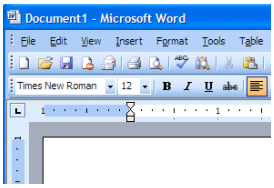


Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognise, diagnose, and recover from errors
- Help and documentation

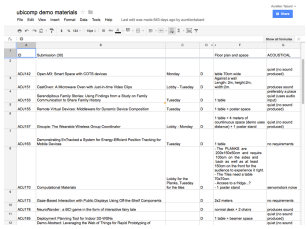
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



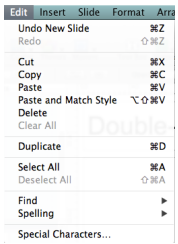
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



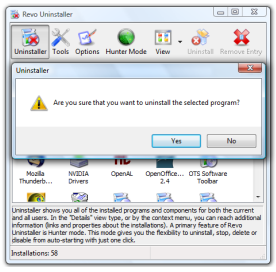
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



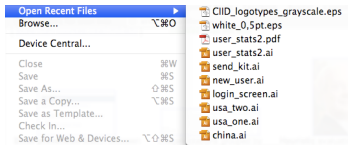
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



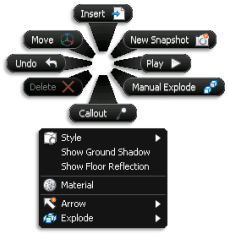
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation

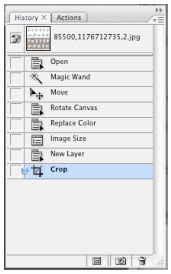


Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation

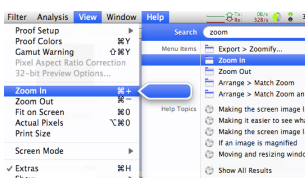
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Échelle de sévérité

Facteurs

- Fréquence : est ce commun?
- Impact: quelle difficulté à résoudre le problème?
- Persistance: faut il affronter le problème dans la durée?

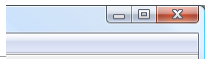
Échelle de sévérité

- Cosmétique : pas nécessaire de le résoudre
- Mineur: résolution nécessaire mais basse priorité
- Majeur: résolution nécessaire et haute priorité
- Catastrophique : résolution obligatoire

Rédiger de bonnes évaluations heuristiques

- Les évaluations heuristiques doivent être compréhensible des développeurs et des manageurs
- Inclure les commentaires positifs et négatifs
  - Ex : La barre d'icône est simple, avec un bon contraste et peu de couleur (minimalist design)
- Avoir du tact
  - Pas : l'organisation des menus est bordélique
  - Plutôt : les menus ne sont pas organisés par fonction
- Être spécifique
  - Pas : le texte est illisible
  - Plutôt : le text est trop petit avec un mauvais contraste (texte noir sur fond vert)

Exemple



Quoi inclure :

- Le problème
- L'heuristique
- La description
- La sévérité
- Des recommendations
- Un screenshot (si ça aide)

Severe: User may close window without saving data (error prevention)

If the user has made changes without saving,
and then closes the window using the Close
button, rather than File >> Exit, no confirmation
dialog appears.

Recommendation: show a confirmation dialog or save automatically

Résumé

- L'évaluation heuristique est une méthode "low-cost"
- Il est mieux que les évaluateurs passe deux fois sur l'interface
  - Leur demander si elle respecte les heuristiques
  - Noter quand ce n'est pas le cas et pourquoi
- Les évaluateurs notent la sévérité indépendamment
- Combiner les découvertes de 3 à 5 evaluateurs
- Discuter les problemes avec l'équipe de design
- Alternative (moins chère) au test utilisateur
- Identifie souvent des problèmes différents alors bonne alternative.

Évaluation et tests

- **Introduction**
- **Approches d'évaluation**
- **Méthodes analytiques**

Plus tard dans le cours :

- Évaluation 2.0 : passer à l'échelle

Activité : Design Walkthrough

Évaluation analytique formative informelle

Évaluer un aspect précis pas à pas

- l'interface
- un scénario
- un prototype
- ...

Comme pour le brainstorming :

- implication de divers participants (designers, développeurs, marketing...)
- temps limité
- recueillir le maximum de commentaires
- les critiques viennent avec des suggestions (= être constructif)