

Ergonomie des IHM

Module IHM



Évaluation heuristique et
Cognitive Walkthrough

Plan

- ◆ **Evaluation heuristique**
- ◆ Cognitive Walkthrough

Évaluation heuristique

- ◆ **Promenade (*walkthrough*) informelle**
 - A partir d'une liste de principes (*heuristics*)
 - inspecter l'interface
 - afin d'identifier des problèmes potentiels d'utilisabilité

Auteur : Jacob Nielsen, [http:// www.useit.com](http://www.useit.com)

Évaluation heuristique

◆ Avantages et inconvénients

- Economique et facile à appliquer (+)
 - Aucun utilisateur n'est requis
 - Peut être réalisée individuellement
 - Nécessite une planification minimale
- Utilisable très tôt dans le cycle de conception (+)
- Problèmes liés au domaine de l'application difficilement identifiables (-)

« *E.H. = a discount usability engineering method* »

Évaluation heuristique

◆ Méthode

- Un groupe d'évaluateurs (plus on est, plus on trouve des problèmes)
- 3-5 personnes
- Des « heuristiques » de conception facilitent l'inspection

Évaluation heuristique

◆ Heuristiques initiales

- Dialogue simple et naturel
- Parler le langage des utilisateurs
- Ne pas surcharger la mémoire des utilisateurs
- Cohérence
- Feed-back
- Sorties clairement indiquées
- Raccourcis
- Messages d'erreur appropriés
- Prévenir les erreurs
- Aide et documentation

Évaluation heuristique

◆ Dialogue simple et naturel

- La manière d'interagir avec le système doit être évidente
- Regrouper les objets de l'interface
- Le nombre d'informations à l'écran doit être suffisant
- Vue générale et détails
- Eviter les « décorations » qui perturbent

Évaluation heuristique

◆ Parler le langage des utilisateurs

- Utiliser le langage du domaine
- Eviter le langage informatique
- Utiliser des métaphores familières
- Utiliser des icônes, images, etc., en rapport avec le domaine

Évaluation heuristique

- ◆ **Ne pas surcharger la mémoire des utilisateurs**
 - Fournir suffisamment d'information à l'écran
 - Privilégier la reconnaissance plutôt que le rappel
 - Mettre en valeur l'information importante
 - Fournir des valeurs par défaut

Évaluation heuristique

◆ Cohérence

- Cohérence de la disposition des informations a l'écran (*layout*)
- Cohérence de l'interaction
- Cohérence du langage
- Cohérence des fonctionnalités

Évaluation heuristique

◆ **Feed-back**

- Indiquer quand le système est en train de faire un traitement
- Fournir un feed-back aux actions de l'utilisateur
- Fournir un feed-back en rapport avec la tâche de l'utilisateur
- Indiquer quand le système a échoué

Évaluation heuristique

◆ **Sorties clairement indiquées**

- Montrer le chemin suivi par l'utilisateur dans le système
- Montrer comment aller en avant et en arrière
- Montrer comment s'orienter dans l'espace des informations

Évaluation heuristique

◆ Raccourcis

- Utiliser des raccourcis pour réaliser les actions
- Inférer la suite d'une frappe clavier
- Autoriser les sauts vers les endroits où l'utilisateur veut aller
- Réutiliser l'historique de l'interaction

Évaluation heuristique

- ◆ **Messages d'erreurs pertinents**
 - Eviter d'utiliser un langage codé
Exemple : « Erreur No 31 »
 - Utiliser le langage des utilisateurs

Évaluation heuristique

◆ Prévenir les erreurs

- Engager un dialogue système-utilisateur lorsque des actions de l'utilisateur peuvent conduire à des conséquences désastreuses
- Indiquer le statut du système et de ses actions
- Utiliser les commandes « *undo* » et « *redo* »

Évaluation heuristique

◆ Aide et documentation

- Les manuels papier ne devraient pas être nécessaires
- Fournir l'aide en ligne à la demande de l'utilisateur
- Sinon, fournir l'aide à l'initiative du système

Évaluation heuristique

◆ Estimation de la gravité des problèmes

0 = Ceci n'est pas un problème d'utilisabilité

1 = Problème « cosmétique » uniquement : pas nécessaire de le régler si l'on n'a pas le temps

2 = Problème mineur : problème à régler avec une priorité faible

3 = Problème majeur : important à régler (priorité élevée)

4 = Problème « catastrophique » : à régler impérativement avant de diffuser l'interface

Évaluation heuristique

◆ Heuristiques révisées

Nielsen, http://www.useit.com/papers/heuristic/heuristic_list.html

- Visibilité du statut du système
- Compatibilité entre système et monde réel
- Contrôle par l'utilisateur et liberté de l'utilisateur
- Cohérence et normes
- Prévention des erreurs
- Reconnaissance plutôt que rappel
- Flexibilité et efficacité d'utilisation
- Esthétique et conception minimaliste
- Aider les utilisateurs à reconnaître, diagnostiquer et réparer les erreurs
- Aide et documentation

Évaluation heuristique

- ◆ **Autres critères possibles** pour une « évaluation heuristique étendue »
 - **Principes élémentaires de conception d'interface** (Bruce Tognazzini)
<http://www.asktog.com/basics/firstPrinciples.html>
 - **Principes de conception** (C. Marlin «Lin» Brown)
<http://www.inria.fr/acacia/ESSI/Principes.html>
 - **Propriétés d'utilisabilité** (cf. cours de L. Nigay)
 - **Critères ergonomiques** (Scapin et Bastien)
<http://www.crim.ca/~arbastie/criteres.html>

Évaluation heuristique

◆ Heuristiques pour collecticiels

Baker, K., Greenberg, S. and Gutwin, C. (2001) **Heuristic Evaluation of Groupware Based on the Mechanics of Collaboration.**

In M.R. Little and L. Nigay (Eds) Engineering for Human-Computer Interaction (8th IFIP International Conference, EHCI 2001, Toronto, Canada, May), Lecture Notes in Computer Science Vol 2254, p123-139, Springer-Verlag. Revised from Report 2000-669-21.

<http://www.cpsc.ucalgary.ca/grouplab/papers/2001/01-HeuristicsMechanics.EHCI/01-Heuristic%20Evaluation%20of%20Groupware.EHCI.pdf>

Évaluation heuristique

◆ Heuristiques pour collecticiels

- Provide the means for intentional and appropriate verbal communication
- Provide the means for intentional and appropriate gestural communication
- Provide consequential communication of an individual 's embodiment
- Provide consequential communication of shared artifacts (i.e. artifact feedthrough)
- Provide protection
- Management of tightly and loosely-coupled collaboration
- Allow people to coordinate their actions
- Facilitate finding collaborators and establishing contact

Évaluation heuristique

◆ Participatory Heuristic Evaluation

- Evaluation heuristique réalisée avec des utilisateurs
- Heuristiques supplémentaires (orientées processus)
 - Respecter l'utilisateur et ses « habiletés »
 - Fournir une expérience plaisante avec le système
 - Assister la réalisation d'un travail (produit) de qualité

Plan

- ◆ Evaluation heuristique
- ◆ **Cognitive Walkthrough**

Cognitive Walkthrough :

« Cheminement/Promenade cognitive »

- ◆ Un évaluateur « chemine » à travers l'interface
- ◆ Il essaie d'agir comme un utilisateur
- ◆ Ses actions sont basées sur un modèle générique de l'activité d'un utilisateur
 - Rapide (+)
 - Grossière (*dirty*) (-)
 - N'implique pas des utilisateurs (-)
 - Ne permet pas d'identifier des problèmes liés au domaine (-)

Cognitive Walkthrough

◆ **Technique d'évaluation**

centrée sur la capacité de l'interface à faciliter le processus d'« apprentissage exploratoire » chez l'utilisateur

= possibilité pour l'utilisateur d'utiliser l'interface sans formation initiale approfondie

Cognitive Walkthrough

- ◆ Méthode basée sur la « théorie cognitive de l'apprentissage initial » de Lewis & Polson
 - = Modèle de la cognition humaine décrivant l'interaction H-M en quatre étapes

Cognitive Walkthrough

◆ Les 4 étapes de l'interaction H-M

- 1) L'utilisateur **se donne** un **objectif** à réaliser à l'aide du système (ex. : vérifier l'orthographe d'un document)
- 2) L'utilisateur **recherche** dans l'interface les **actions** qu'il peut réaliser (items de menu, boutons, commandes clavier, etc.)
- 3) L'utilisateur **choisit l'action** la plus appropriée pour atteindre le but recherché
- 4) L'utilisateur **réalise l'action** et **évalue le feed-back** du système en fonction de l'objectif à atteindre

Cognitive Walkthrough

◆ Procédure

- 1) Pré-requis du « cheminement »
- 2) « Cheminement » (en fonction des 4 étapes)

Cognitive Walkthrough

◆ Pré-requis du « cheminement »

- a) Description générale des **utilisateurs** et de leurs **connaissances**
- b) Description spécifique d'une ou plusieurs **tâches** représentatives à réaliser à l'aide du système — **scénarios d'utilisation**
- c) Liste des actions correctes nécessaires pour réaliser ces tâches — **séquences d'actions**

Cognitive Walkthrough

◆ Cheminement

- Passer en revue chaque étape de la séquence d'action correcte et
- essayer de « raconter une histoire crédible expliquant pourquoi le futur utilisateur de l'interface devrait choisir les actions prévues »

Cognitive Walkthrough

- ◆ **Pour évaluer** la facilité avec laquelle l'action correcte sera choisie (ou la crédibilité des « histoires »), l'évaluateur peut prendre en considération **4 critères** :
 - le but de l'utilisateur
 - l'accessibilité de la « commande » correcte
 - le degré de compatibilité entre le « nom de la commande » et le but de l'utilisateur
 - le feed-back fourni après l'activation de la commande

Cognitive Walkthrough

◆ Etapes de l'interaction IHM et questions correspondantes

■ Se donner une action à réaliser

(l'utilisateur sait-il quoi faire ?)

■ Réaliser l'action

(l'utilisateur sait-il comment faire?)

■ Interpréter les résultats de l'action

(l'utilisateur comprend-il ce que le système vient de faire ?)

Cognitive Walkthrough

- ◆ **Variante de la méthode : walkthrough réalisé par un groupe de concepteurs**
 - L'un des évaluateurs joue le rôle de « **scribe** », et enregistre au fur et à mesure les résultats de l'évaluation
 - Un autre évaluateur joue le rôle de « **facilitateur** »

 - Des walkthroughs « à petite échelle » de parties de l'interface peuvent être réalisés par un concepteur unique

Cognitive Walkthrough

◆ Exemple d'utilisation

David G. Novick (1999). Using the cognitive walkthrough for operating procedures, *Interactions*, 6(3), 31-37.

<http://www.cs.utep.edu/novick/papers/cw-op.interactions.html>

Cognitive Walkthrough

*Interface
à évaluer*

Procedure:	Respond to a request to report		
Sequence:	000	Author:	David G. Novick
Version:	10	Date:	05 Nov 1998

- Why
- What
- How

Goals	Acts	Actions
Delay response due to transient cockpit workload.	PNF • If you need a short-term delay to respond to the clearance, respond STBY.	Select DCDU LSK for STBY.
Ensure mutual awareness.	PNF • Announce: "Responding standby."	Read aloud from DCDU.
Inform ATC that the aircraft will comply with the request to report.	PNF • Respond ROGER.	Select DCDU LSK for ROGER.
Ensure mutual awareness.	PNF • Announce: "Responding roger to request to report."	Read aloud from DCDU.

Find
Mode Edit
 Add

Update

Cognitive Walkthrough

*Exemple
d'une page
de
couverture*

Cognitive Walkthrough for Operating Procedures Cover Sheet

Date: 07/08/98

Analyst(s): PB, PR, MC, LM, SS, DN

Users: Airline pilots, trained on this model aircraft

Interface: Air traffic control cockpit interface, plus flight management system.

Task:

- (1) Procedure: Respond to a clearance
- (2) Procedure: Respond to a request to report
- (3) Technique: Respond

Action sequence: As specified in the documentation, plus comparison with industry-standard specification if necessary

Comments: (1) The "build" language is not very helpful

Cognitive Walkthrough for Operating Procedures

Story: Success Failure

Date: 07/08/98

Analyst: DN

Task: Respond

Step: build-the-message

Walkthrough

1. Will the users try to achieve the intended effect?

Yes, explicit instruction

2. Will the users notice that the correct action is available? No

Experience

a. Documentation: No indication of how to "build" — I/F has choices but build is general description for all

b. Interface: Target msgs will be suggested by the datalink interface in some cases, but not in others.

3. Will the users associate the correct action with the effect trying to be achieved?

No

4. If the correct action is performed, will the users see that progress is being made toward solution of the task?

Yes

Observations

1. Are experience or training needed? Yes.

If so,

a. Is this kind of step common or rare?

b. Will training be easy or difficult?

2. Is the step correct in terms of function?

No. Proc should be modified to avoid "build".

Make the step conditional: "if suggested message has to be modified..."

3. Are particular errors likely?

If so, what is their impact on safety?

4. Design suggestions

Cover cases of "building" messages explicitly in the manual's Description section

5. Other comments

Exemple d'une
« histoire »
relative à une
étape de la
procédure