

# INTRODUCTION

L'évaluation est une phase primordiale dans la boucle de conception des IHM. Elle permet de vérifier et valider un système interactif.

Un système est dit vérifié s'il correspond aux spécifications issues de l'analyse des besoins.

Un système est dit validé s'il correspond aux besoins tout en respectant les contraintes du domaine d'application.

# ERGONOMIE - DÉFINITIONS

## Ergonomie

- vise à adapter les postes de travail aux caractéristiques physiologiques et psychologiques de la personne

## Ergonomie logicielle

- vise à adapter les logiciels à l'utilisateur pour diminuer les erreurs, le temps d'apprentissage
- rendre le logiciel le plus facilement utilisable

# ÉVALUER, QUOI !!!

- Fiabilité et qualité technique
  - logiciel
- Utilisabilité
  - capacité à permettre à l'utilisateur d'atteindre facilement ses objectifs
  - qualité de l'interface (ergonomie)
- Utilité
  - adéquation aux objectifs de haut niveau du client
  - le logiciel satisfait-il les spécifications ?
- Usages
  - utilisation réelle du logiciel
  - le logiciel est-il utilisé comme prévu ?
- Mais la qualité d'un logiciel est perçue comme un tout

# ÉVALUER, QUAND

## En cours de conception

- conception itérative de maquettes/prototypes
- conception centrée utilisateur
- évaluation formative

## En cours de réalisation

- méthodes classiques de contrôle qualité

## En cours de diffusion

- satisfaction des utilisateurs
- incidents critiques

## Avant un achat

- comparaison de logiciels
- évaluation sommative

# ÉVALUER L'UTILITÉ D'UN SYSTÈME

Adéquation aux objectifs de haut niveau du client

- le logiciel satisfait-il les spécifications
- l'utilisateur peut-il accomplir sa tâche à partir des fonctionnalités du système
  - nécessité de bien formaliser la tâche

# ÉVALUER L'UTILISABILITÉ D'UN SYSTÈME

Utilisabilité : norme ISO 9241

- degré selon lequel un produit peut être utilisé, par des utilisateurs identifiés, pour atteindre des buts définis avec efficacité, efficience et satisfaction, dans un contexte d'utilisation spécifié

Critères de l'utilisabilité

- efficacité: atteinte du résultat prévu
- efficience: consommation d'un minimum de ressources
- satisfaction de l'utilisateur: confort et évaluation subjective
- facilité d'apprentissage
- facilité d'appropriation: prise en main du logiciel
- facilité d'utilisation
- fiabilité: pas ou peu d'erreurs d'utilisation

# TECHNIQUES D'ÉVALUATION DE L'UTILISABILITÉ

## Évaluation expérimentale

- observation et recueil de données
- entretiens, questionnaires

## Évaluation analytique

- scénario d'utilisation, jugements d'experts
- utilisation de référents
  - évaluations heuristiques
  - critères d'évaluation ergonomique

# JUGEMENTS D'EXPERTS

## Caractéristiques

- nécessité d'avoir plusieurs experts
- en début de projet pour évaluer des produits existants
- utiles pour éliminer les grosses erreurs
- nécessité de disposer de scénarios d'utilisation du système et d'une description des futurs utilisateurs

## Mais...

- les experts ne peuvent pas remplacer complètement les utilisateurs
- les experts ont souvent des domaines privilégiés qui constituent des biais d'évaluation

# QU'EST-CE QU'UNE ANALYSE HEURISTIQUE ?

- L'analyse heuristique consiste à inspecter une interface en la comparant à une liste de critères, les « heuristiques », de manière à détecter des aspects positifs ou négatifs du point de vue de l'utilisabilité. Ces critères permettent notamment de catégoriser les défauts d'ergonomie, de juger de leur importance et de trouver des solutions pour les résoudre. Il ne s'agit pas d'évaluer l'adéquation des fonctionnalités avec les besoins des utilisateurs, ni d'étudier les interactions avec le système, mais bien de constater des écarts par rapport aux standards décrits par les heuristiques.

# CRITÈRES D'ÉVALUATION ERGONOMIQUE

Il existe plusieurs grilles d'analyse :

- Les critères de Bastien et Scapin
- Les heuristiques de Nielsen
- Les règles d'or de Coutaz
- Les golden rules de Schneiderman
- Les critères de Tognazzini
- Etc ...



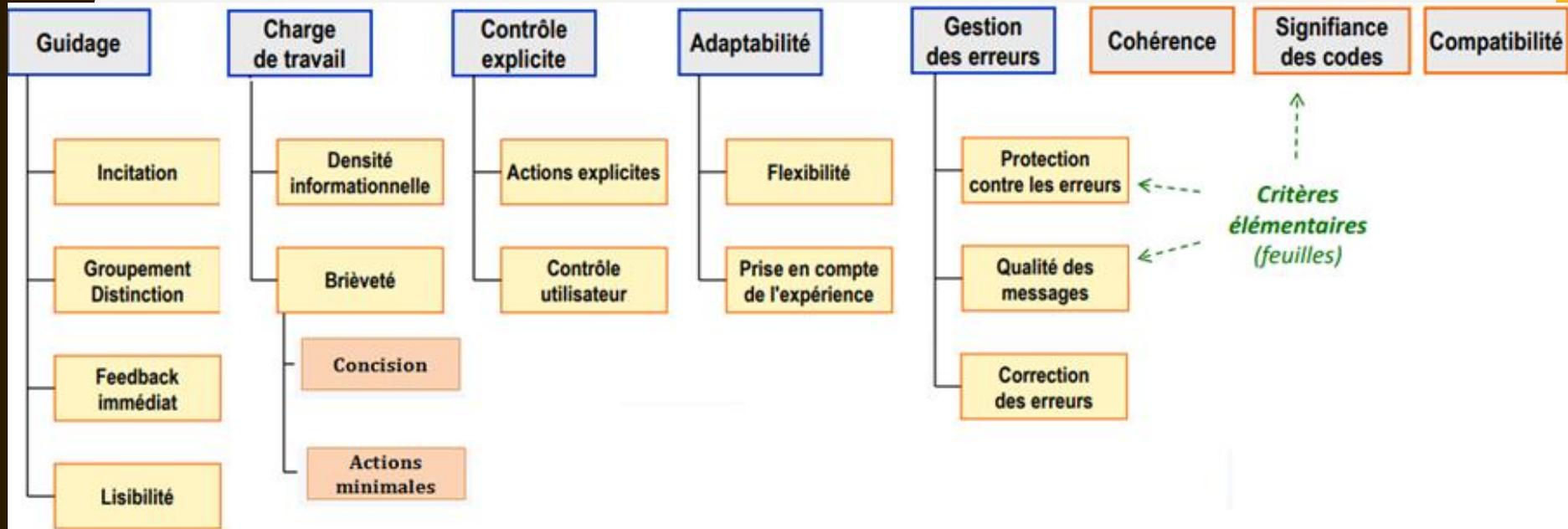
***CRITÈRES ERGONOMIQUES  
DE BASTIEN ET SCAPIN***

# ***CRITÈRES ERGONOMIQUES DE BASTIEN ET SCAPIN***

- Deux chercheurs français qui ont procédé à partir de 1997 à la synthèse de 900 recommandations dans le domaine de l'ergonomie informatique (pris dans son sens le plus large).
- Leur travail a abouti à une liste de 8 critères principaux et 18 sous critères .
- Les études ont montré que leur grille révélait davantage de défauts d'interface que les autres listes (et notamment celle de la norme ISO 9241 10).

# CRITÈRES ERGONOMIQUES DE BASTIEN ET SCAPIN

## Critères principaux



# 1. LE GUIDAGE

- Ensemble des moyens mis en œuvre pour conseiller, orienter, informer et conduire l'utilisateur lors de ses interactions avec l'ordinateur.
- L'utilisateur doit comprendre clairement ce qu'il peut faire et comment il peut le faire. Le système doit donc le prendre en main pour le conduire efficacement.
- On distingue
  - ✓ Le guidage explicite (par ex. messages, bulles d'aide, boîtes de dialogue, ...)
  - ✓ Le guidage implicite (par ex. griser les fonctions inactives, autres artifices, ...)
- Le critère de guidage est décomposé en quatre sous-critères :
  - **Incitation**
  - **Groupement / Distinction**
  - **Feedback immédiat**
  - **Lisibilité**

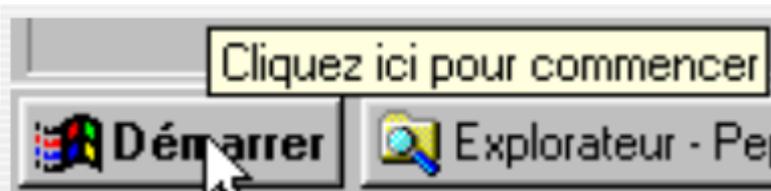
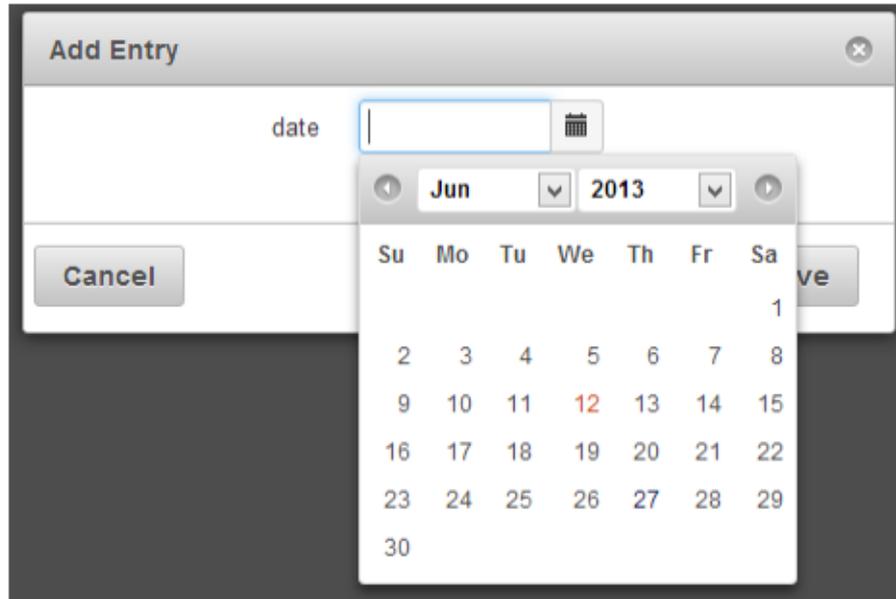
# 1.1 INCITATION

- Inciter l'utilisateur à effectuer des actions spécifiques en **lui fournissant des indices**. Par exemple, guider les entrées de données en indiquant le format adéquat et les valeurs acceptables: Date (jj/mm/aa) \_\_ / \_\_ / \_\_.

## Recommandations :

- Griser les fonctions non disponibles (options de menu, boutons, ...)
- Fournir la liste des saisies attendues (listes déroulantes, codes à utiliser, ...)
- Donner le format de saisie des données (dates, dimensions, ...)
- Modifier la forme du curseur (pointeur de la souris) pour donner des indications sur l'opération à effectuer
- Indiquer clairement les champs obligatoires (\* ou autre indicateur)
- Montrer clairement comment aller en avant et en arrière
- Afficher des bulles d'aides (*Tooltips*) sur les éléments non-triviaux (icônes des barres d'outils par exemple)

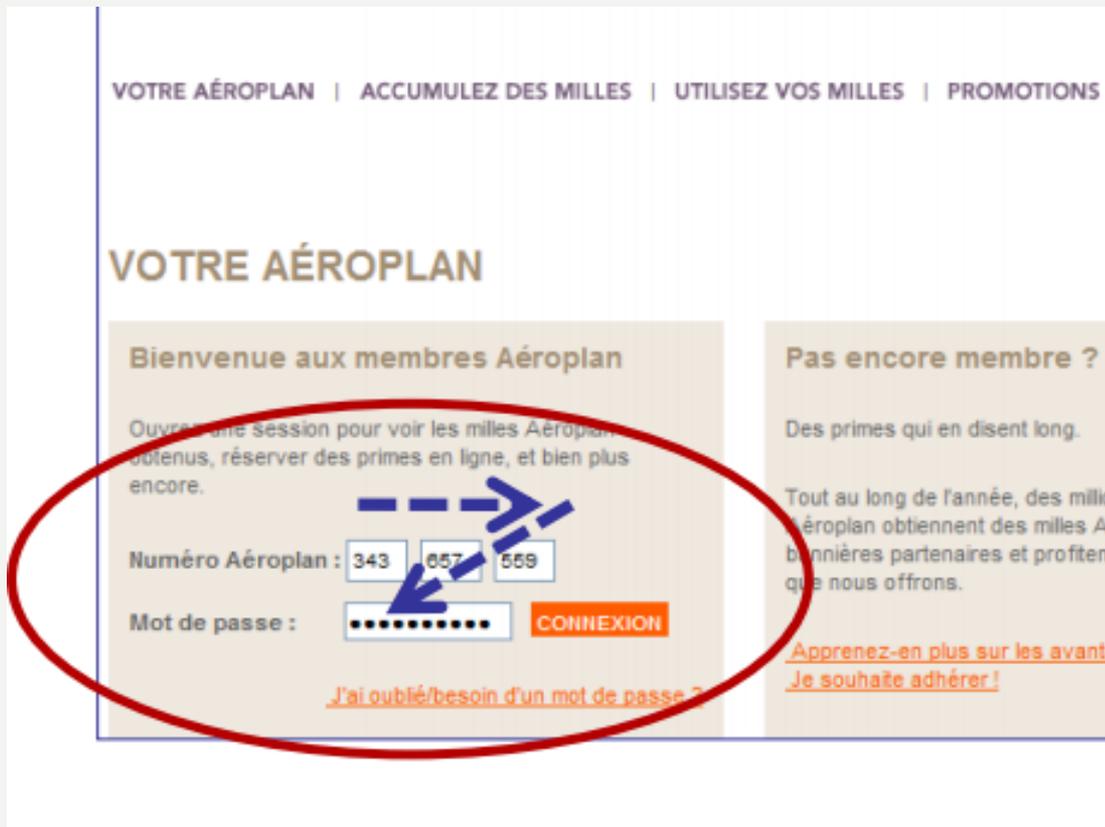
# 1.1. INCITATION : EXEMPLE



# 1.1. INCITATION : EXEMPLE

Un exemple d'incitation réussie :

- Le numéro d'abonné est divisé en trois blocs de trois chiffres
- Une fois un premier bloc saisi, le curseur se déplace automatiquement au bloc suivant



VOTRE AÉROPLAN | ACCUMULEZ DES MILLES | UTILISEZ VOS MILLES | PROMOTIONS

## VOTRE AÉROPLAN

**Bienvenue aux membres Aéroplan**

Ouvrez une session pour voir les milles Aéroplan obtenus, réserver des primes en ligne, et bien plus encore.

Numéro Aéroplan :

Mot de passe :

[J'ai oublié/besoin d'un mot de passe ?](#)

**Pas encore membre ?**

Des primes qui en disent long.

Tout au long de l'année, des milliers d'Aéroplan obtiennent des milles Aéroplan à travers nos partenaires et profitent de nos offres.

[Apprenez-en plus sur les avantages](#)

[Je souhaite adhérer !](#)

# 1.1. INCITATION : CONTRE EXEMPLE

Un exemple où l'incitation est si mauvaise que l'on doit explicitement dire à l'utilisateur où il doit cliquer !



The screenshot shows a website navigation menu on the left and a main content area on the right. The navigation menu includes a 'Home' link circled in red, and other links like 'Introduction', 'Contacting Us', 'Other Nuffield projects', and 'Salter's curriculum Projects (York)'. The main content area features a large image of mushrooms. A red arrow points from the 'Home' link to the mushrooms. Below the mushrooms, the text reads 'Welcome' followed by 'Failure!' in red. Below that, it says 'Please use hexagons on left of mushrooms to navigate.' and 'This web site is for you to find out about the Salters-Nuffield Advanced Biology Project which started in'.

Home

- ▶ Introduction
- Contacting Us
- ▶ Other Nuffield projects
- ▶ Salter's curriculum Projects (York )

**Welcome**

**Failure!**

Please use hexagons on left of mushrooms to navigate.

This web site is for you to find out about the Salters-Nuffield Advanced Biology Project which started in

# 1.2. GROUPEMENT

## DISTINCTION ENTRE ITEMS

- Grouper les différents éléments visuels de façon cohérente et ordonnée. L'objectif est que les caractéristiques visuelles de l'interface nous renseigne sur le sens.
- Deux attributs graphiques sont utilisés dans ce but :
  - **Le format** de présentation (couleur, forme, syntaxe, ...)
  - **La position** dans l'interface (localisation, bordure, ...)

# 1.2. GROUPEMENT

## DISTINCTION ENTRE ITEMS

- A faire

Erreurs à l'exécution	Options de syntaxe
<input type="checkbox"/> Vérification <u>L</u> imites	<input type="checkbox"/> Chaînes-var <u>s</u> trictes
<input type="checkbox"/> Vérification <u>P</u> ile	<input type="checkbox"/> Eval. booléenne <u>c</u> omplète
<input type="checkbox"/> Vérification <u>E/S</u>	<input checked="" type="checkbox"/> Syntaxe <u>é</u> tendue
<input type="checkbox"/> <u>V</u> érification Débordements	<input type="checkbox"/> Opérateur <u>@</u> typé
	<input type="checkbox"/> Paramètres <u>o</u> uverts

- A ne pas faire

Reverse  
Silence  
Invert  
Music...  
Amplify...  
Channel Mixer...  
Compressor...  
Delay...  
Distortion...  
Echo...  
Echo Chamber...  
Envelope...  
Filter...  
Flanger...  
Noise Reduction...  
Normalize...  
Quick Filter...  
Reverb...  
Activer Windows  
Stretch...  
Accédez aux paramètres  
Wave...Windows.

# 1.3. FEEDBACK IMMÉDIAT

- Il s'agit d'informer l'utilisateur sur l'efficacité de ses actions. Dans tous les cas, l'ordinateur doit répondre à l'utilisateur en fonction des actions et des requêtes de ce dernier. (exemple: timer).
- Le système doit réagir à chacune des actions de l'utilisateur et ce quelque soit la longueur du traitement (long ou non).

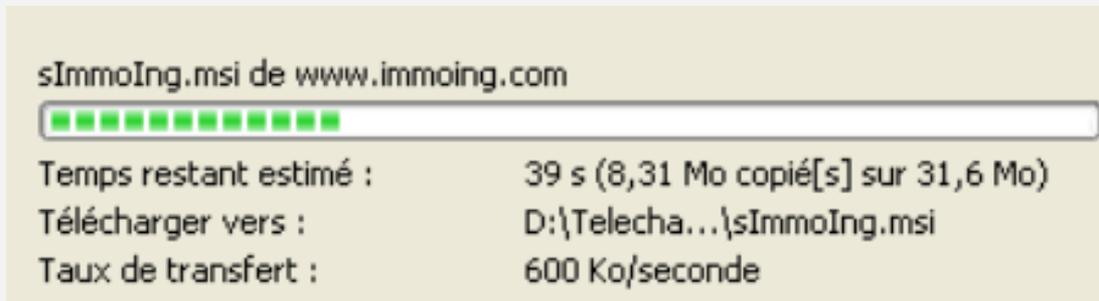
## **Recommandations :**

- Le logiciel doit répondre à toute action de l'utilisateur par un changement dans la présentation de l'interface (visuel, sonore, ...)
- Indiquer les modes de fonctionnement du système (état)
- Signaler les traitements longs par une indication d'attente (sablier, barre de progression, animation, message, ...)
- Toujours faire apparaître les saisies utilisateur
- Rendre visible les traitements réalisés par le logiciel et indiquer clairement lorsqu'ils ont échoué

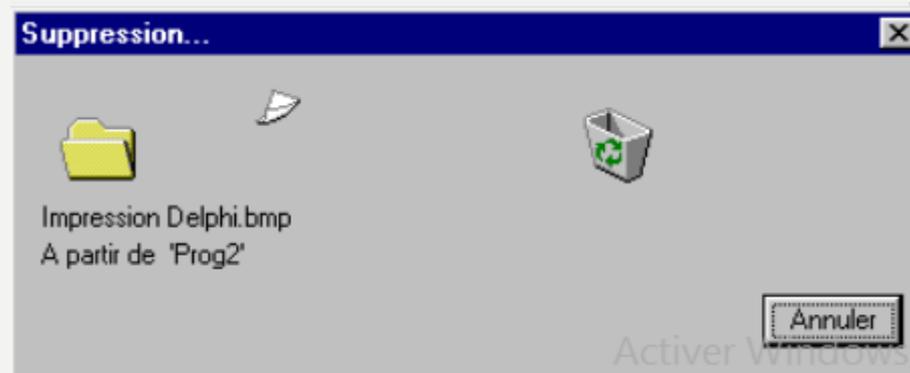
# 1.3. FEEDBACK IMMÉDIAT : EXEMPLE

## A faire :

- changer l'apparence d'un objet **sélectionné**
- changement de curseur pour les attentes courtes
- indicateur de progression pour les tâches plus longues (>5s)



## A ne pas faire



# 1.4. LISIBILITÉ

- • Le critère de lisibilité consiste à faciliter la perception des informations textuelles et iconographiques par un choix judicieux de leurs propriétés et de leur disposition.
- Une attention particulière doit être apportée :
  - Aux **polices** de caractères (diversité, taille, attributs, minuscules/majuscules, ...)
  - Aux **couleurs** en général (choix, contraste, but, ...)
  - À la **disposition** des éléments (alignements, espacements, mise en page,

## Recommandations :

- Utiliser une police de caractère lisible (à l'écran, éviter les italiques en petite taille, ...)
- Soigner les espacements (interligne) et les alignements (justification)
- Adapter la taille des libellés et des icônes afin de garantir leur interprétation
- Soigner les contrastes (choix des couleurs pour le fond et le premier plan)
- Prendre en compte la configuration de la machine cible (résolution, ...)

# 1.4. LISIBILITÉ: EXEMPLE ET CONTRE EXEMPLE

- A faire : contraste

Ce texte est plus lisible : noir sur fond blanc, caractères sans empattement

- A ne pas faire

Ce texte est plus difficile à lire : mauvaise association de couleurs, caractères avec empattement

# 2. CHARGE DE TRAVAIL

- Le critère de charge de travail regroupe l'ensemble des moyens visant à réduire la charge perceptive, mnésique et physique de l'utilisateur.
- On doit donc viser à minimiser à la fois la quantité d'informations que l'utilisateur doit prendre en compte ainsi que le nombre d'actions élémentaires qu'il doit réaliser pour accomplir une tâche donnée.
- Le critère de la charge de travail se décompose en deux sous-critères élémentaires :
  - **Brièveté**
  - **Densité informationnelle**

# 2.1. BRIÈVETÉ

- Limiter le travail de lecture, d'entrée et les étapes par lesquelles doivent passer les usagers
- De plus, on doit faciliter le traitement cognitif des informations. La taille des groupes comme la taille des éléments individuels (mots, boutons, liens...) doivent donc être adaptés.

## Recommandations :

- Limiter le nombre d'options dans un menu ou dans une liste déroulante
- Éviter les libellés trop longs
- Réduire le nombre d'actions élémentaires pour atteindre un objectif donné
- Éviter à l'utilisateur d'avoir à se souvenir d'informations d'une fenêtre à l'autre, d'avoir à faire des calculs ou de saisir des informations qui peuvent être déduites par le système.

# 2.1. BRIÈVETÉ

- Quelles sont les fonctions nécessaires, utiles (l'abondance nuit dans certaines circonstances).



# 2.1.1. CONCISION

- Réduire la charge de travail au niveau perceptif et mnésique (perception et mémorisation) pour ce qui est des éléments individuels d'entrée ou de sortie.
- Exemple

Exemple : **taille du libellé d'un lien** :

Pour nous contacter, vous pouvez cliquer sur ce lien et alors on vous répondra très vite

versus

contactez-nous

# 2.1.1. CONCISION

- A faire :

Lancer

- A ne pas faire

Lancement de l'application de gestion des notes

Tapez ici le numéro d'identification de l'étudiant dont vous voulez saisir les notes

## 2.1.2. ACTIONS MINIMALES

- Limiter les étapes par lesquelles doivent passer les utilisateurs.
- Par exemple, ne pas demander aux utilisateurs d'entrer des données qui peuvent être déduites par le système.

### Contre Exemple:

**GUICHET AUTOMATIQUE EN SERVICE**

Insérer votre carte

Entrer votre code d'identification : **1234**

Sélectionner une opération : **retrait**

Donner le montant du retrait : **7500 F**

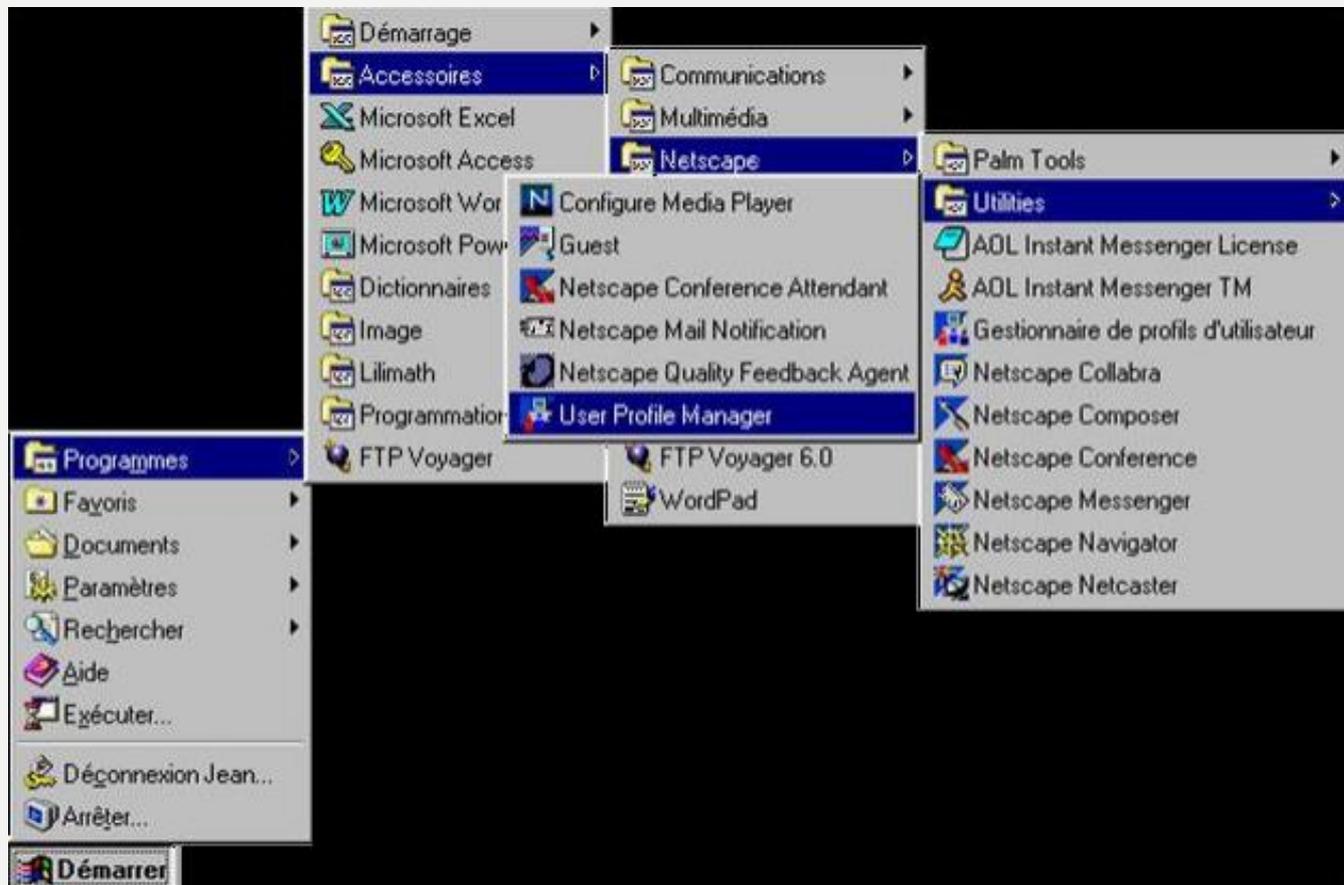
Le montant doit être un multiple de **1000 F**

(La machine éjecte la carte)

**GUICHET AUTOMATIQUE EN SERVICE...**

(L'utilisateur doit tout recommencer)

# 2.1.2. ACTIONS MINIMALES: A NE PAS FAIRE



# 2.2. DENSITÉ INFORMATIONNELLE

- Réduire la charge de travail du point de vue perceptif et mnésique, pour des ensembles d'éléments et non pour des items (ce qui doit être perçu et mémorisé globalement par l'utilisateur).
- Par exemple, limiter la densité informationnelle de l'écran, en affichant seulement les informations nécessaires.

## Recommandations :

- N'afficher que les informations pertinentes pour effectuer la tâche (boîtes de dialogue simples, représentations graphiques, ...)
- Éviter les écrans trop chargés (décomposer si nécessaire)
- Éviter les liens trop nombreux dans un texte affiché sur une page web
- Éviter les textes trop verbeux (dialogue simple, phrases courtes)
- Privilégier la reconnaissance (symboles, icônes)

# 2.2. DENSITÉ INFORMATIONNELLE

Richesse du contenu, mais certainement pas idéal.

**Resto-ranG** Genève

Bern Basel St. Genève Lausanne Zürich

recherche groupes restaurateurs

Nom  cuisine  lieu  catégorie

**Le site n°1 des horaires ciné**

home | aide | contact | Login | DE EN

**top20 petit budget [?]**

1	O Sole Mio	6.59
2	Olio & Salvia	6.56
3	La Caravane passe...	6.55
4	Bigoudi 13	6.53
5	Café d'Avusy	6.53
6	Les secrets de Philomène	6.44
7	Café Lys	6.43
8	Coq en pâte	6.40
9	tatine.com	6.36
10	Au Royaume de Jade	6.27
11	Le Forum	6.27
12	Auberge de Céligny	6.23
13	Le Bar à Soupe	6.17
14	Le Mosaïque	6.15
15	Le Mouton Noir	6.13
16	Café Fair PLAY	6.11
17	Port Saladin	6.05
18	Mr Pickwick Pub	6.00
19	La Rotonde	5.91
20	Café du Jura	5.90

**top20 budget moyen [?]**

1	Fingers	9.26
2	Le Cazar	8.21
3	La Fontaine	6.63
4	Auberge de Thônex	6.44
5	Lacustre (Bellevue)	6.28
6	le Tariquet	5.95
7	Yupa Porn Pochana	5.90
8	Le Pavillon du Lac	5.86
9	Le Zen	5.82
10	La Ruota	5.60
11	Les Agapes	5.45
12	Malombré	5.43
13	Le Phénix	5.38
14	E.K.O.	5.37
15	L'opéra Bouffe	5.33
16	Auberge Communale de Meyrin	5.31
17	La Ferme	5.25
18	U bobba	5.20
19	Auberge de Vandoeuvres	5.20
20	La Pointe à la Bise	5.18

**top20 chic [?]**

1	Domaine de Châteauevieux	6.34
2	Cittadella	6.28
3	Restaurant de La Capite	6.26
4	Buffet de la gare Eaux-Vives	6.14
5	Le Cigalon	6.12
6	Hostellerie de la Vendée	6.00
7	Le Lavandou	5.94
8	Chez Kei	5.83
9	Giardino Romano	5.83
10	L'Escapade	5.82
11	Jaipur	5.73
12	Aux Trois Bonheurs	5.71
13	Nologo	5.67
14	Le Sénat	5.60
15	Auberge de Landecy	5.59
16	Le Chat Botté	5.57
17	Kotobuki	5.54
18	Na Village	5.53
19	Restaurant du Vallon	5.42
20	Relais Thai de Vuillonex	5.40

**actualités**

NOCTURNES A L'ILE ROUSSEAU - Ile Jean-Jacques Rousseau  
foie gras pœllé aux figues et mangues - Le Pavillon du Lac  
MATÉRIEL D'OCCASION A VENDRE A PRIX CASSÉ - La Pointe à la Bise  
RESTAURANT LE TARIQUET - le Tariquet spécial feu d'artifice - La Broche

[ajoutez une actualité](#) | [toutes les actus](#)

**5 derniers commentaires sur 16414**

**Chez Uchino (Versoix)** - Zarzuella  
Un délice! Un des meilleurs restaurants japonais! Voilà de la bonne cuisine japonaise, et un accueil charmant. Le cadre est très sobre, simple, agréable...[\(suite\)](#)

**cette semaine**

- Le Pavillon du Lac
- Auberge de Thônex
- Olio & Salvia
- Auberge de Landecy
- L'As de Pique
- Le Saladier - Auberge Communale
- Le Portugais
- Le Mortimer

[inscription restaurateurs](#)

**outils**

- [recommander ce site](#)
- [ajouter un resto](#)
- [version mobile](#) m.restorang.ch

**5 derniers restos ajoutés à Genève**

luna	
James Pub	
Via Roma	10.00

HÉBERGÉ PAR **infomaniak NETWORK**

**Promotion Resto 2007**  
Testez nos services pendant un mois sans frais!  
Plus de 50 restaurants nous font déjà confiance...

» Informations  
» OK, appelez-moi!

**Ile Jean-Jacques Rousseau**

*L'Île Rousseau*  
Restaurant Bar Terrasse

L'un des lieux magiques et mythiques de Genève avec un décor unique et un accueil chaleureux Une île - terrasse abritée environnée de Nature et d'oise...[\(suite\)](#)

**l'escalier**

Au rez : un bar en zinc, des couleurs chaudes, un accueil chaleureux A l'étage : notre restaurant 45 places maxi cosy et lumineux grâce à ses grandes ...[\(suite\)](#)

**Restaurant de La Capite**

**Restaurant de La Capite**

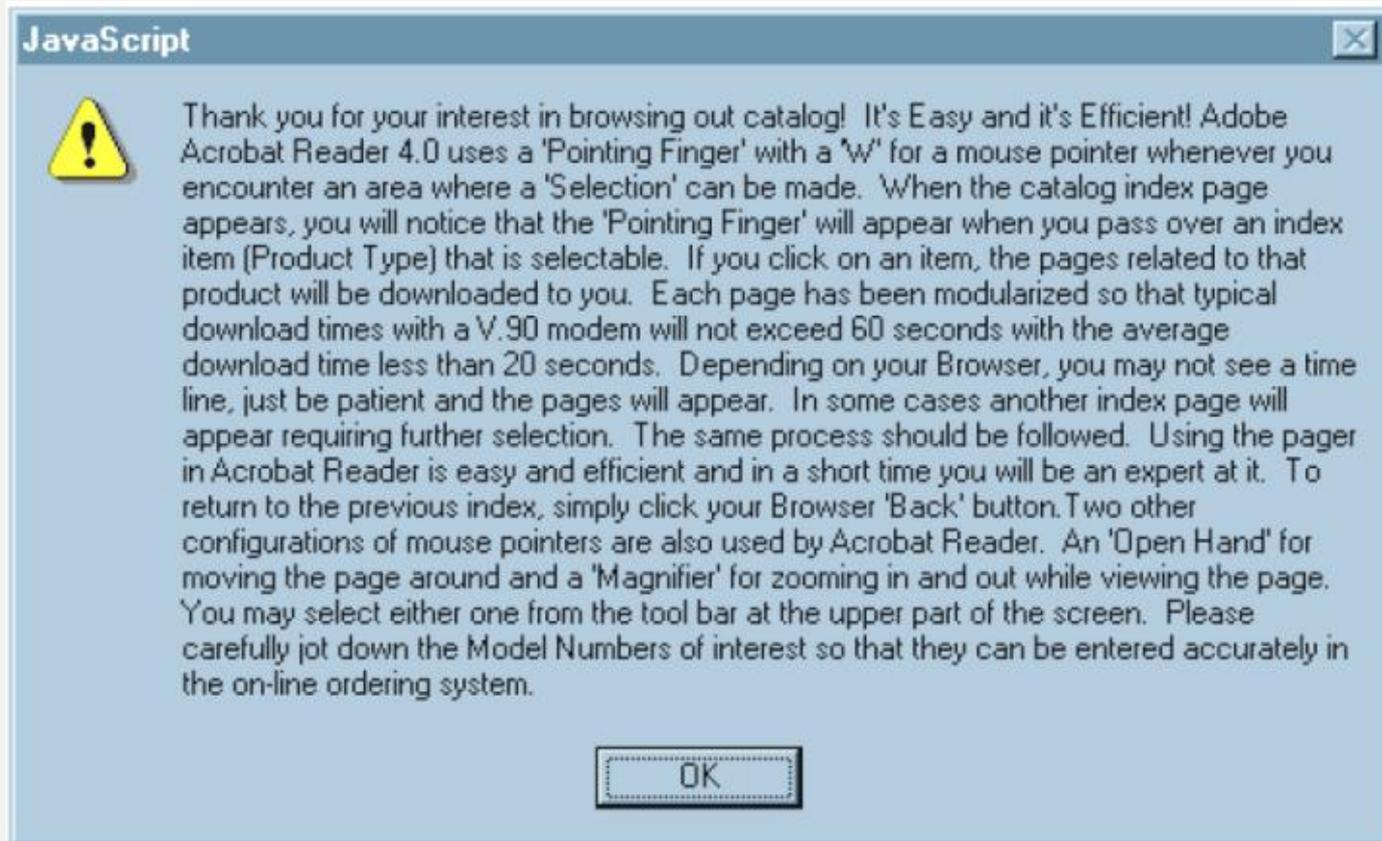
Frédéric et Carine Rossi vous proposent une cuisine ensoleillée et raffinée, réalisée avec des produits frais. Leurs spécialités, selon les saisons : ...[\(suite\)](#)

**Les Fourneaux du Manège**  
*Les Fourneaux*

# 2.2. DENSITÉ INFORMATIONNELLE

Un message que l'on a pas trop envie de lire...

- Éviter les textes trop verbeux (dialogue simple, phrases courtes).



# 3. CONTRÔLE EXPLICITE

- Le critère de contrôle explicite concerne les aspects liés au degré de maîtrise qu'a l'utilisateur sur les traitements réalisés par le système interactif.
- Il se décompose en deux sous-critères élémentaires :
  - **Actions explicites**
  - **Contrôle utilisateur**

# 3.1. ACTIONS EXPLICITES

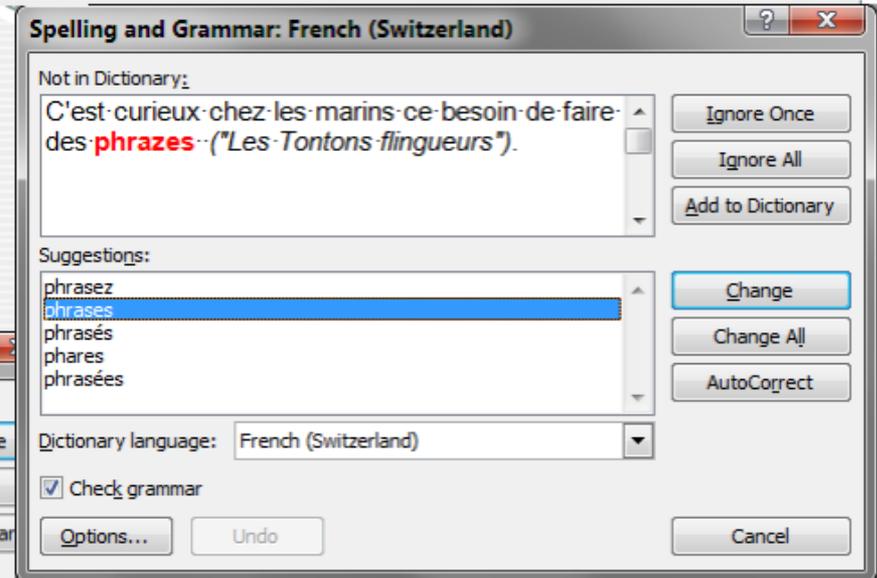
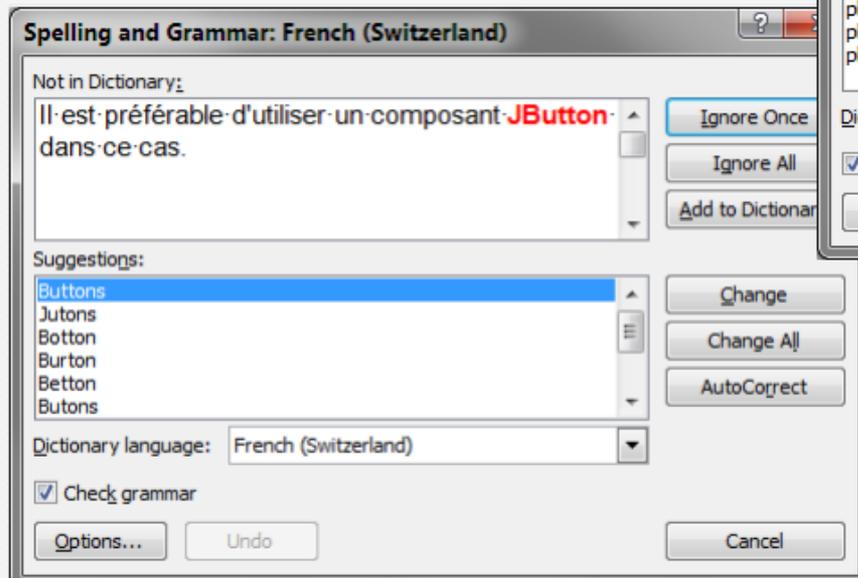
- Expliciter la relation entre le fonctionnement de l'application et les actions des utilisateurs.
- Par exemple, l'entrée de commandes doit se terminer par une indication de fin (« Enter », « OK ») à laquelle des possibilités d'édition doivent être préalables.

## **Recommandations :**

- Ne pas déclencher d'opérations sans le consentement explicite de l'utilisateur.
- Déclencher l'opération immédiatement après l'action de l'utilisateur ou, sinon, indiquer clairement que l'opération sera différée (ou qu'elle ne peut pas être effectuée).

# 3.1. ACTIONS EXPLICITES

- Exemple :
  - Ne pas effectuer de modification dans un document sans demander confirmation à l'utilisateur



## 3.2. CONTRÔLE UTILISATEUR

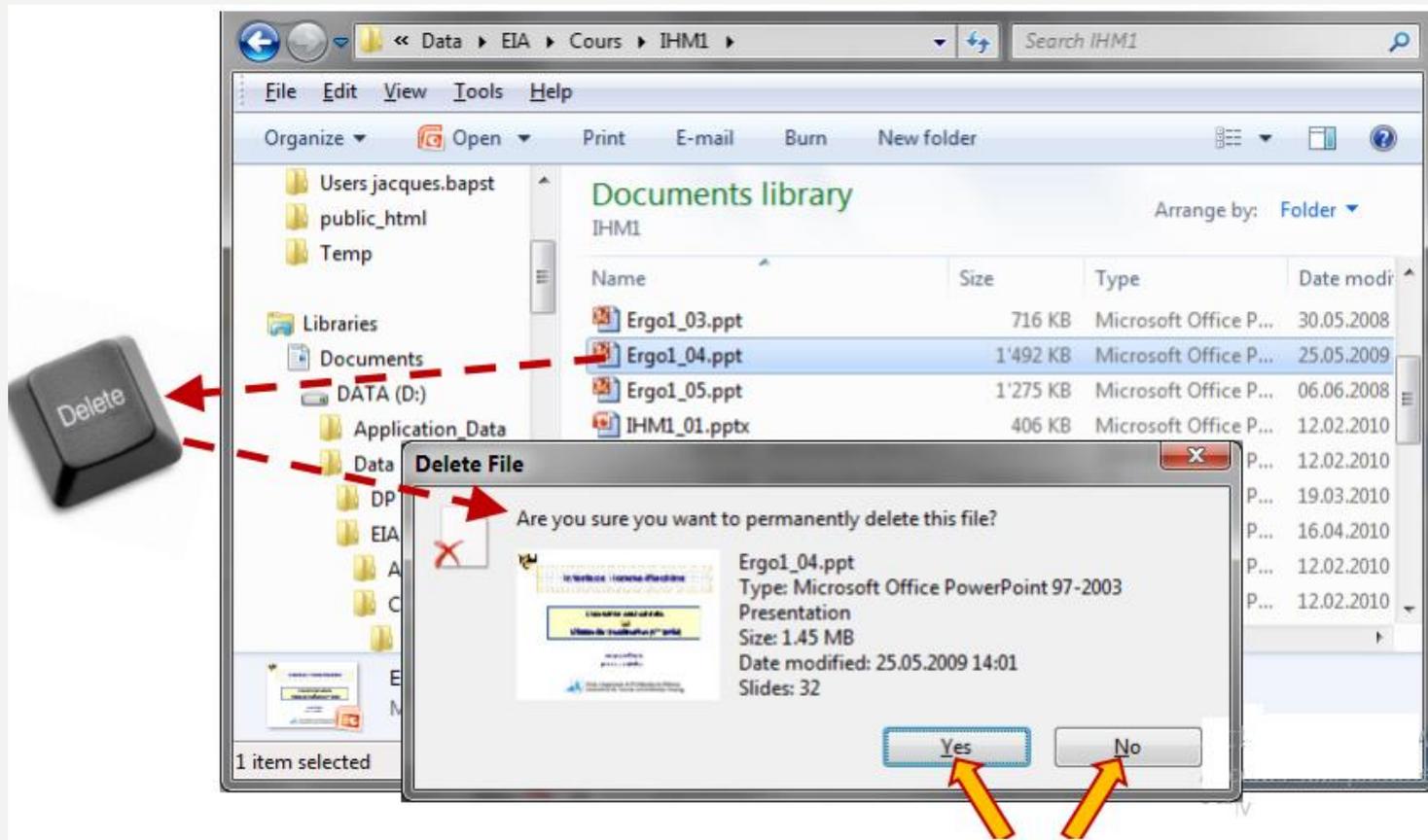
- Le critère de contrôle utilisateur concerne le fait que l'utilisateur doit toujours avoir la main sur le système et en contrôler les opérations et leur déroulement (interrompre, reprendre).
- L'utilisateur doit avoir, en permanence, le contrôle sur le logiciel.

### Recommandations :

- Offrir à l'utilisateur une validation explicite des commandes importantes ou difficilement réversibles
- Offrir la possibilité d'interrompre les traitements longs
- Autoriser les retours en arrière (*Undo*)
- Permettre, en tout temps, de quitter la fonction courante ou même, le logiciel

# 3.2. CONTRÔLE UTILISATEUR: EXEMPLE

- Offrir à l'utilisateur une validation explicite des commandes importantes ou difficilement réversibles



# 3.2. CONTRÔLE UTILISATEUR: CONTRE EXEMPLE

The image shows a screenshot of a software dialog box titled "Identification d'un client". The dialog box has a blue title bar and a white content area. It contains several input fields and buttons:

- Nom :** A text input field.
- Prénom :** A text input field.
- Numéro :** A text input field with a "Vérifier" button to its right.
- Code postal :** A text input field.
- Localité :** A text input field.

On the right side of the dialog box, there are four buttons stacked vertically: "OK", "Annuler", "Aide", and "Valider". The "Valider" button appears to be associated with the "Prénom" and "Localité" fields.

# 4. ADAPTABILITÉ

- L'adaptabilité d'un système est caractérisée par la capacité de son interface à réagir et à s'adapter en fonction du contexte et selon les besoins et les préférences de ses utilisateurs.
- Le critère d'adaptabilité se décompose en deux sous-critères élémentaires :
  - **Flexibilité**
  - **Prise en compte de l'expérience de l'utilisateur**

# 4.1. FLEXIBILITÉ

- Le critère de flexibilité concerne les moyens mis à la disposition des utilisateurs pour personnaliser l'interface afin de tenir compte des préférences, des aptitudes, des habitudes, du contexte d'utilisation et de la manière de travailler de chacun d'eux.
- Il vise également à offrir à l'utilisateur la possibilité d'effectuer une tâche ou activer une fonction de différentes manières, par exemple :
  - Par un menu déroulant
  - Par un menu contextuel
  - Par une icône dans une barre d'outils
  - Par un raccourci clavier

# 4.2. PRISE EN COMPTE DE L'EXPÉRIENCE UTILISATEUR

- Le deuxième sous-critère de la flexibilité est la prise en compte de l'expérience de l'utilisateur qui concerne les moyens mis en œuvre pour s'adapter aux différents niveaux d'expérience des utilisateurs.

## Recommandations :

- Permettre à l'utilisateur de définir son niveau d'expérience.  
Le demander ou le déterminer automatiquement par des métriques (par exemple le nombre ou le taux d'erreurs, les temps de réaction, etc.).
- Guider l'utilisateur débutant (novice) pas à pas dans la réalisation de la tâche.
- Mettre en place un guidage fort en créant des assistants (*wizard*).
- Donner aux utilisateurs expérimentés les moyens d'effectuer leur tâche de manière rapide et efficiente (même si c'est parfois au détriment du guidage).

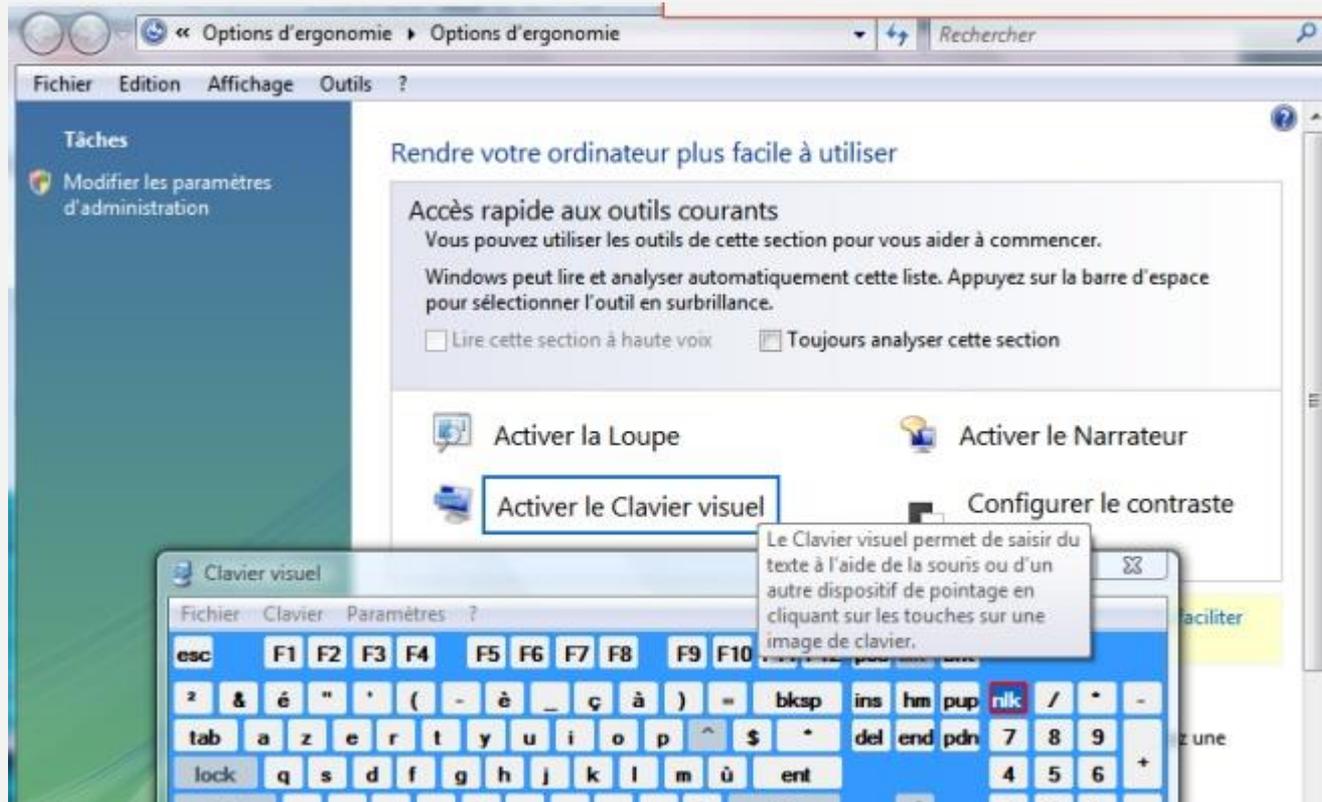
# 4.2. PRISE EN COMPTE DE L'EXPÉRIENCE UTILISATEUR: EXEMPLE

- Cette interface permet à un utilisateur de rechercher un mot clé selon 3 modalités.



The screenshot displays the RERO catalogue search interface. The header features the RERO logo and the text 'catalogue collectif'. A navigation menu on the left includes options like 'Recherche', 'Simple', 'Avancée', 'Espace personnel', 'Panier (0)', 'Historique des recherches', 'Catalogues RERO', 'Autres ressources', and 'Services en ligne'. The main content area is titled 'Recherche avancée' and offers three search modalities: 'Par mot-clé', 'Par sujet', and 'En mode expert'. A search input field is present, with an example query 'a:nothomb & t:tubes' provided below it. A 'Rechercher' button is located at the bottom right of the search area.

# 4.2. PRISE EN COMPTE DE L'EXPÉRIENCE UTILISATEUR: EXEMPLE



# 5. GESTION DES ERREURS

- Moyens permettant d'une part d'éviter ou de réduire les erreurs, d'autre part de les corriger lorsqu'elles surviennent.
- L'objectif prioritaire est de minimiser les interruptions dues aux erreurs (maximiser la performance, l'efficacité, la productivité).
- Trois sous-critères sont à prendre en compte :
  - Protection contre les erreurs (éviter que l'utilisateur n'en commette)
  - Qualité des messages d'erreur (informer clairement l'utilisateur)
  - Correction des erreurs (lui permettre de les corriger)

## Recommandations générales :

- Tout mettre en œuvre pour éviter les erreurs (attitude défensive)
- Engager un dialogue lorsque des actions de l'utilisateur peuvent conduire à des situations irréversibles (ou des conséquences fâcheuses)
- Avertir l'utilisateur au plus tôt et le guider vers la résolution du problème
- Faciliter l'exploration et l'apprentissage du système

# 5.1. PROTECTION DES ERREURS

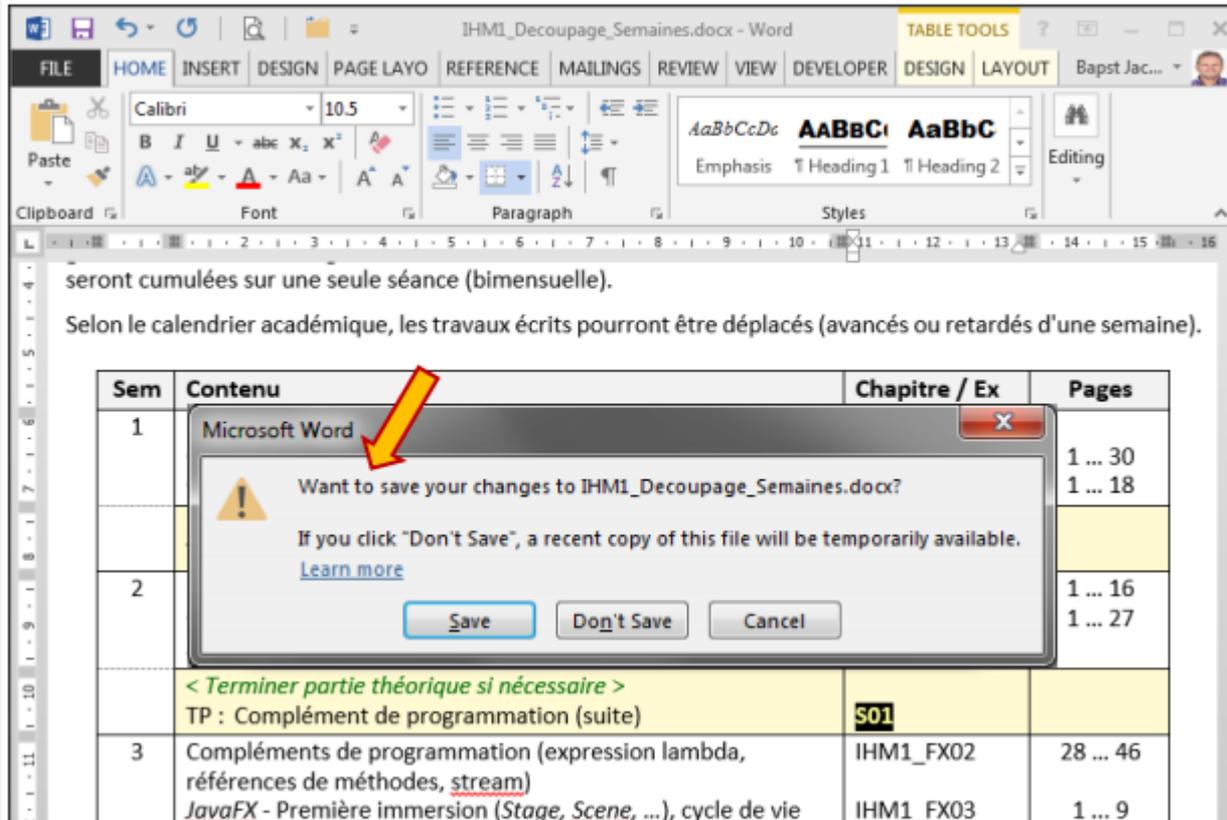
- Mettre en place des moyens pour détecter et prévenir les erreurs (avant validation).
- Si une erreur survient malgré tout, il faut la détecter au plus tôt et avertir l'utilisateur avec un message clair.
- Par exemple, toutes les actions possibles sur une interface doivent être envisagées et plus particulièrement les appuis accidentels des touches du clavier afin que les entrées non-attendues soient détectées.

## Recommandations :

- Mettre en évidence les commandes non disponibles (griser) *Prévention*
- Fournir la liste des valeurs possibles, des unités, ... *Prévention*
- Détecter les erreurs au plus tôt et avertir immédiatement l'utilisateur
- Minimiser les saisies au clavier (si possible, listes à choix)
- Prévenir les risques de perte de données (demander confirmation)

# 5.1. PROTECTION DES ERREURS: EXEMPLE

- Prévenir les risques de perte de données (demander confirmation)

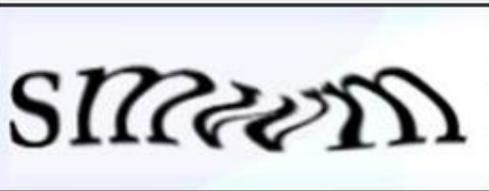


The screenshot shows the Microsoft Word interface with a 'Save' dialog box open. The dialog box asks: 'Want to save your changes to IHM1\_Decoupage\_Semaines.docx?' and includes a warning icon, a 'Learn more' link, and buttons for 'Save', 'Don't Save', and 'Cancel'. A red arrow points to the dialog box. The background shows a table of contents with columns for 'Sem', 'Contenu', 'Chapitre / Ex', and 'Pages'.

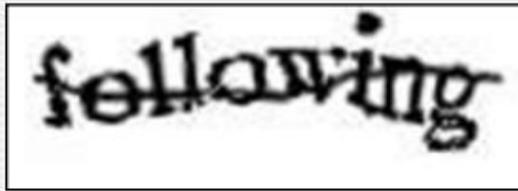
Sem	Contenu	Chapitre / Ex	Pages
1	Microsoft Word		1 ... 30 1 ... 18
2			1 ... 16 1 ... 27
	< Terminer partie théorique si nécessaire > TP : Complément de programmation (suite)	S01	
3	Compléments de programmation (expression lambda, références de méthodes, <u>stream</u> ) <u>JavaFX</u> - Première immersion ( <u>Stage</u> , <u>Scene</u> , ...), cycle de vie	IHM1_FX02 IHM1_FX03	28 ... 46 1 ... 9

# 5.1. PROTECTION DES ERREURS: EXEMPLE

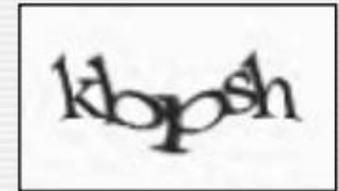
- **A faire :**
  - Captcha / tips : permettant de différencier de manière automatisée un utilisateur humain d'un ordinateur



Distorsion des lettres et ajout d'un dégradé de couleur en fond



reCAPTCHA : ajout d'une ligne brisée



Imbrication des lettres les unes dans les autres (CAPTCHA actuel de Yahoo).

Answer?

# 5.2. QUALITÉ DES MESSAGES D'ERREURS

- Le critère de qualité des messages d'erreur concerne la pertinence, la facilité de lecture et l'exactitude des informations données aux utilisateurs sur la nature des erreurs commises ainsi que sur les actions à entreprendre pour les corriger.
- Pour que la correction de l'erreur soit aisée, le message d'erreur doit indiquer la nature de l'erreur, sa cause, ainsi que les moyens de la corriger (la qualité des messages favorise l'apprentissage du système).
- Le contenu des messages doit tendre à dédramatiser les erreurs aux yeux de l'utilisateur (diminue son sentiment de culpabilité et le stress

## **Recommandations :**

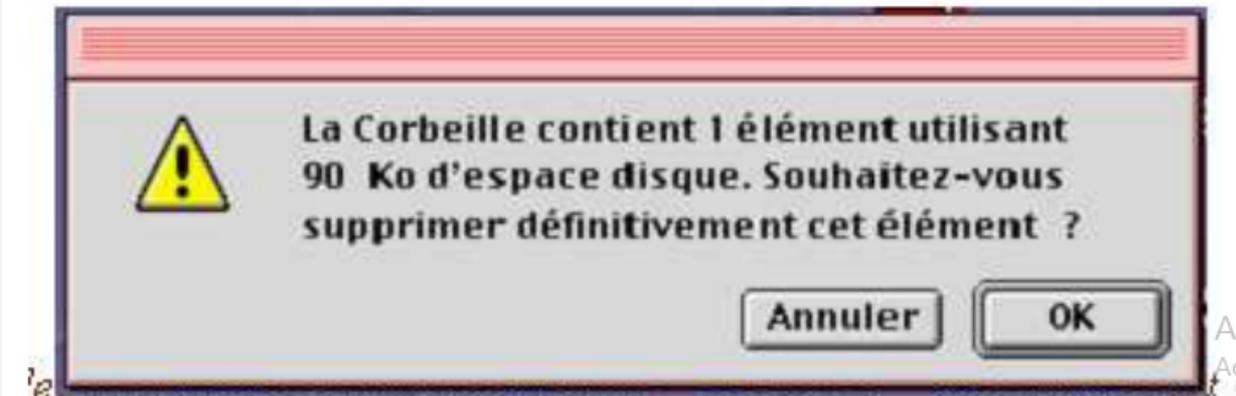
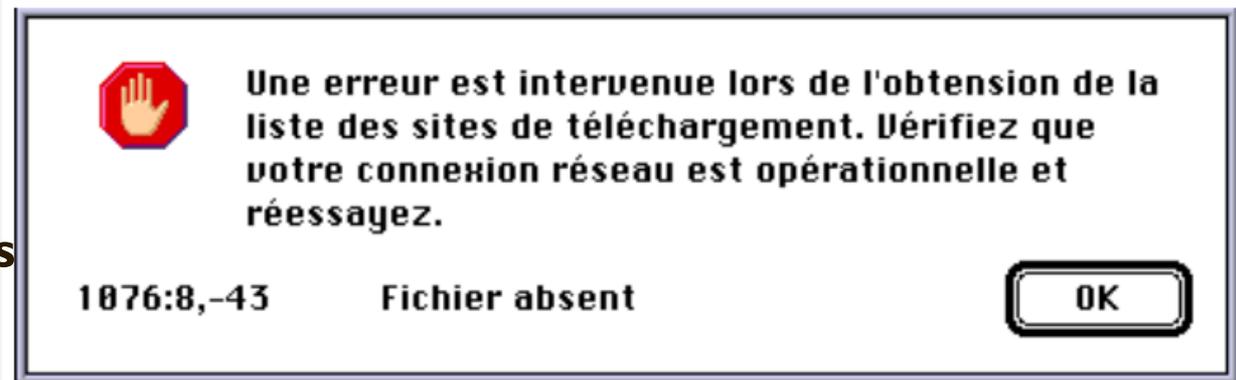
- Placer les messages d'erreur là où l'utilisateur est censé regarder
- Afficher des messages d'erreur explicites (utiliser le langage de l'utilisateur)
- Éviter les textes trop longs (rester bref, utiliser des liens, des références, ...)
- Éviter les textes réprobateurs
- Faire en sorte, si possible, que les textes soient auto-suffisants. Trouver le bon compromis entre concision et exhaustivité

# 5.2. QUALITÉ DES MESSAGES D'ERREURS: EXEMPLE

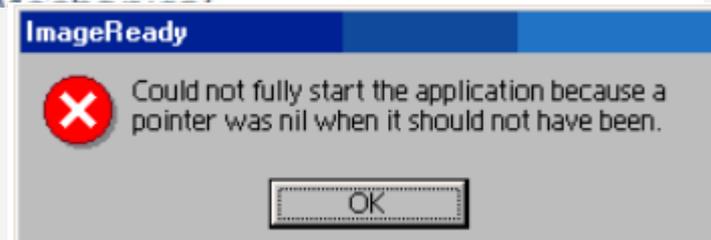
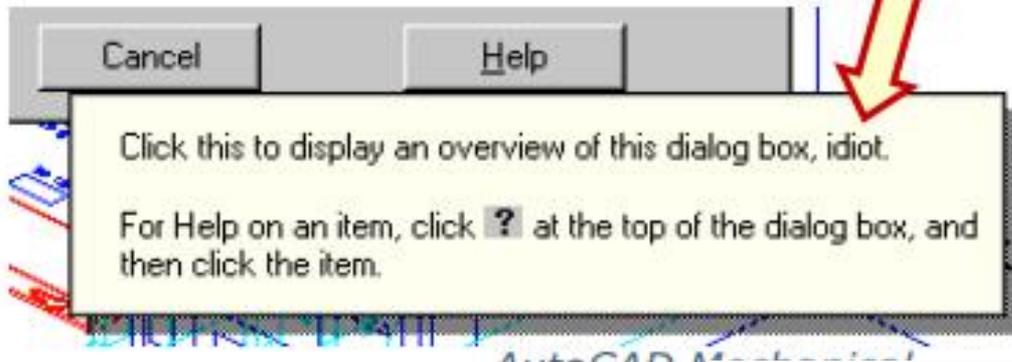
messages informatifs, constructifs, compréhensibles, amicaux, montrant la cause et la solution

## 4 types de messages

- information
- avertissement
- erreur bloquante
- indication de progression



# EXEMPLES À NE PAS SUIVRE !

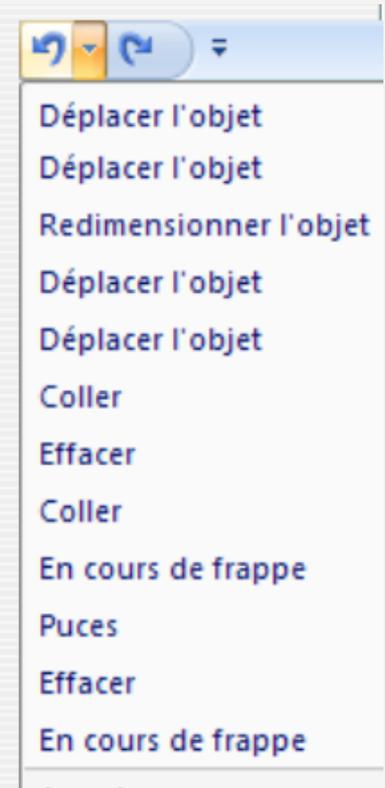


# 5.3. CORRECTION DES ERREURS

- Le critère de correction des erreurs inclut tous les moyens mis à la disposition des utilisateurs pour leur permettre de corriger leurs erreurs.
- Offrir à l'utilisateur les moyens de corriger le plus facilement et le plus rapidement possible ses erreurs en lui permettant, par exemple, de ne corriger que la portion des données qui est erronée et ne pas le contraindre à ressaisir l'ensemble des informations.
- Tactiques différentes selon le type et la gravité de l'erreur :
  - Bloquer l'utilisateur tant que l'erreur subsiste (erreurs graves)
  - Lui permettre de continuer après une mise en garde (message, signal sonore)
  - Ne pas répondre à la commande erronée (un message est nécessaire)
  - Corriger automatiquement l'erreur (dans quelques rares cas).

# 5.3. CORRECTION DES ERREURS

- A faire :
  - la commande annuler
  - messages explicites



# 5.3. CORRECTION DES ERREURS

- A faire :
  - la vérification des saisies
  - Mettre en évidence le champ/élément erroné

Blog URL

Email  Email taken or invalid. If you have an account [sign in](#).

Password

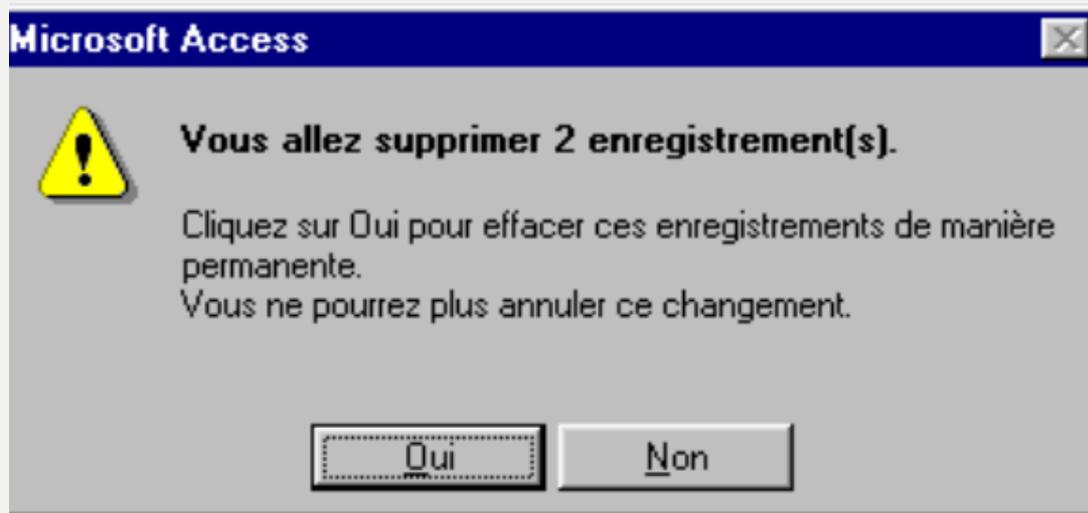
Display Name

Gender  Male  Female  Decline to state

Birthday

# 5.3. CORRECTION DES ERREURS

- A ne pas faire :



# 6. HOMOGÉNÉITÉ / COHÉRENCE

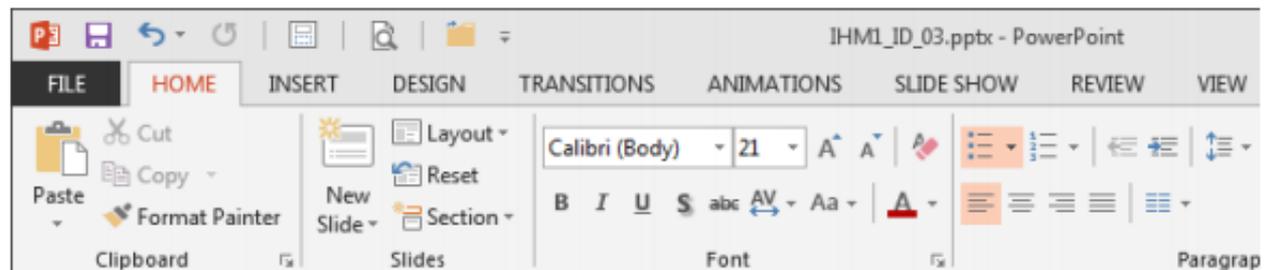
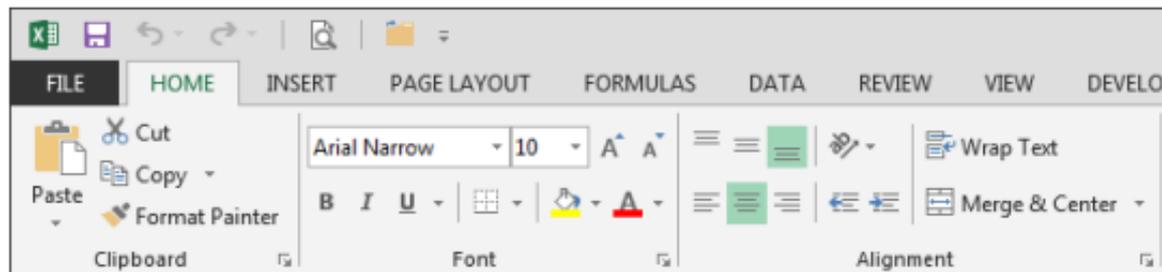
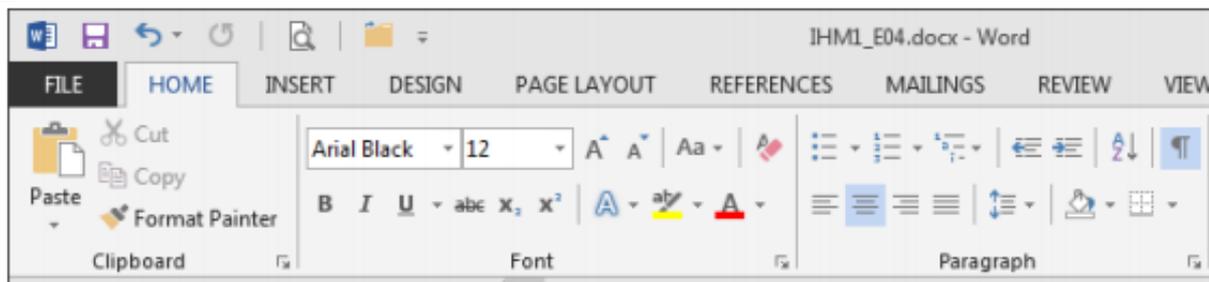
- ❑ Le critère de cohérence concerne l'homogénéité globale de l'interface homme-machine.
  
- ❑ L'objectif est de respecter une logique cohérente pour :
  - La présentation (graphisme, localisation, vocabulaire, format, syntaxe, ...)
  - Le comportement (réaction du système, messages, retours sonores, ...)

## Recommandations :

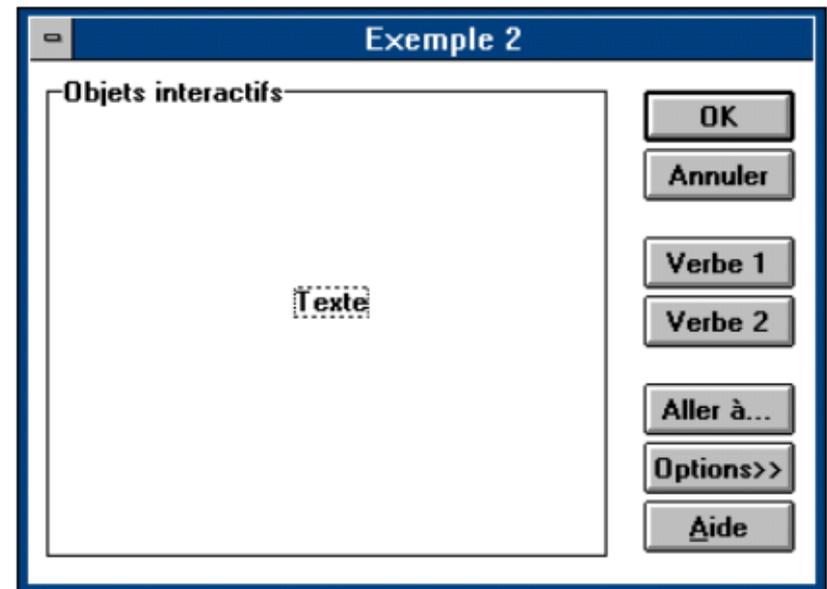
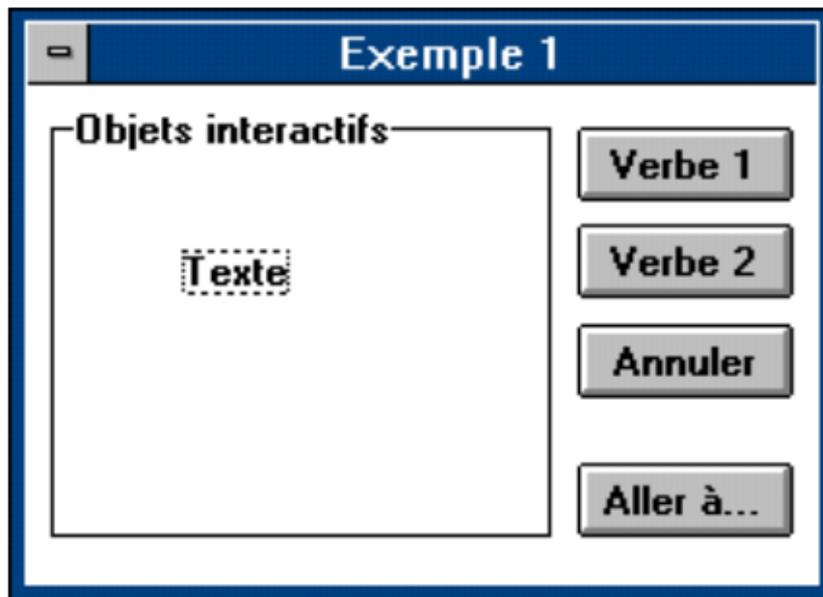
- Utiliser le même schéma d'agencement pour toutes les fenêtres (gabarit d'écran appelé aussi *tracé régulateur*)
- La sémantique des boutons de la souris doit être constante
- Le même vocabulaire doit être utilisé pour désigner les mêmes fonctions
- Utiliser une organisation et une syntaxe cohérente pour les menus
- Utiliser de manière cohérente les symboles graphiques (icônes, couleurs, Acces)

# 6. HOMOGENÉITÉ / COHÉRENCE: EXEMPLE

Microsoft Office, Word, Excel, PowerPoint : une certaine cohérence dans les menus (mais qui pourrait être encore largement améliorée).

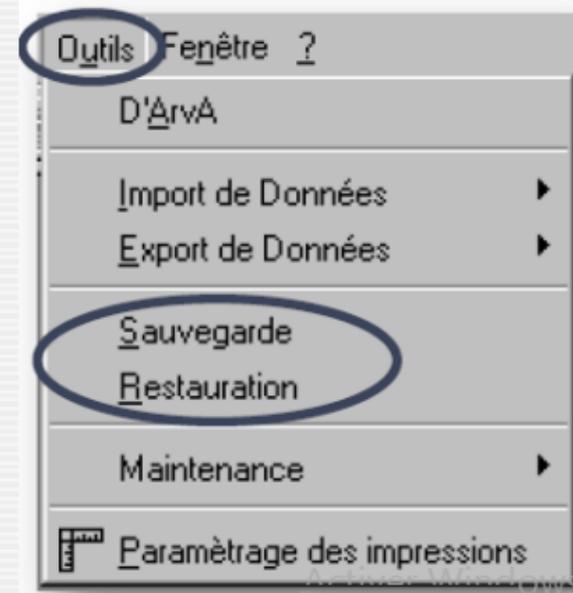
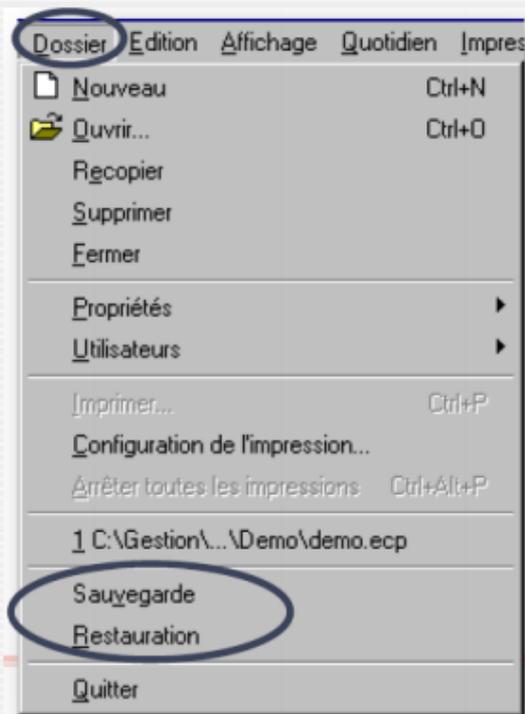
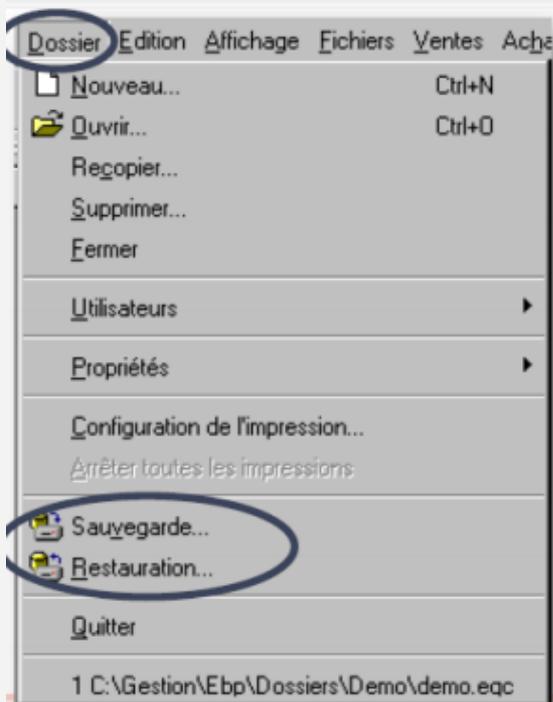


# 6. HOMOGÉNÉITÉ / COHÉRENCE



# 6. HOMOGENÉITÉ / COHÉRENCE: CONTRE EXEMPLE

» A ne pas faire :



# 7. SIGNIFIANCE DES CODES ET DÉNOMINATIONS

- La signifiante des codes et dénominations caractérise l'adéquation entre l'objet, l'information ou le comportement présentés par l'interface et son référent (l'objet ou l'action qu'il représente).
- Il doit y avoir adéquation entre l'objet ou l'information affichée ou entrée, et son référent. Autrement dit, il ne doit pas y avoir d'ambiguïté dans le sens donné aux mots ou aux icônes (et plus globalement à l'ensemble du langage)

## **Recommandations :**

- Éviter les termes techniques (jargon), parler le langage de l'utilisateur
- Reproduire le comportement habituel des objets (par ex. la gomme)
- Définir explicitement et respecter les règles d'abréviation
- Prendre en compte les standards en vigueur (standards formels ou de facto) pour toutes les dénominations

# 7. SIGNIFIANCE DES CODES ET DÉNOMINATIONS: EXEMPLES



Billet de loterie virtuel que l'on gratte avec le curseur.

# 7. SIGNIFIANCE DES CODES ET DÉNOMINATIONS: CONTRE EXEMPLES

**International - Format de la date**

**Format abrégé**

Ordre:  MJA  JMA  AMJ

Séparateur:

Zéro non significatif pour le jour

Zéro non significatif pour le mois

Siècle sur quatre chiffres

**Format complet**

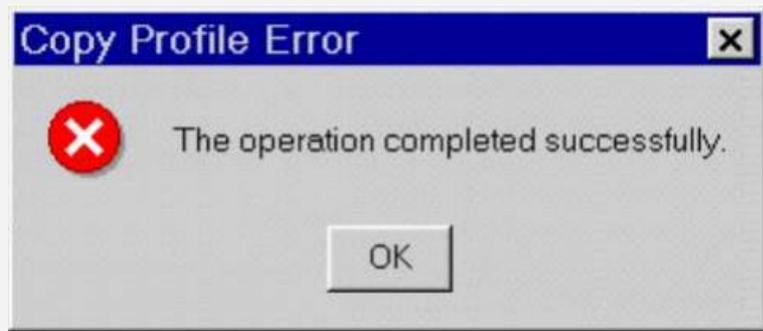
Ordre:  MJA  JMA  AMJ

lundi 5 mars 1991

lundi 5 mars 1991

OK

Annuler



# 8. COMPATIBILITÉ

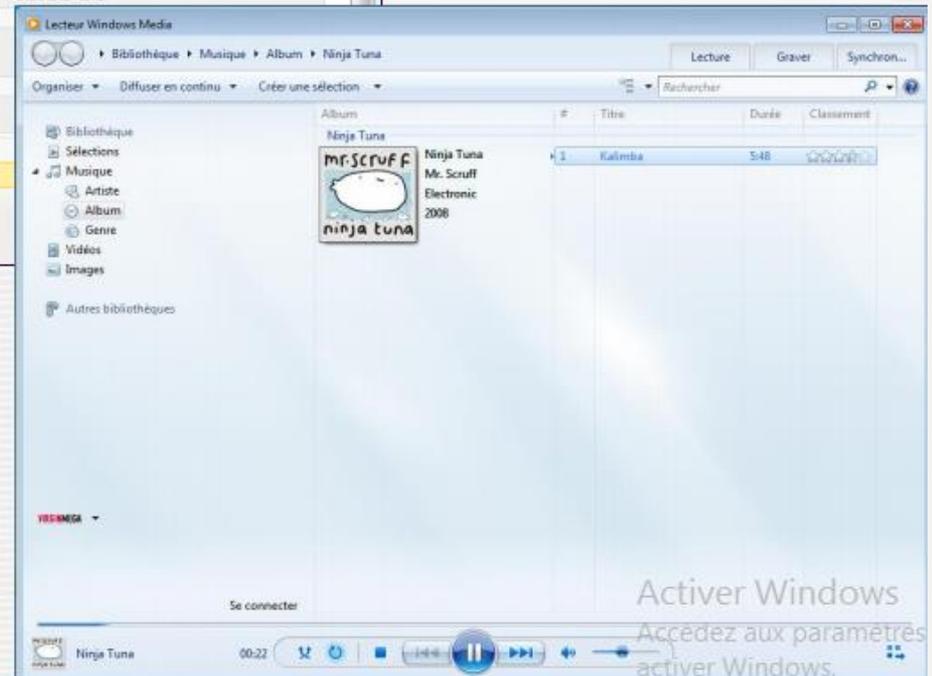
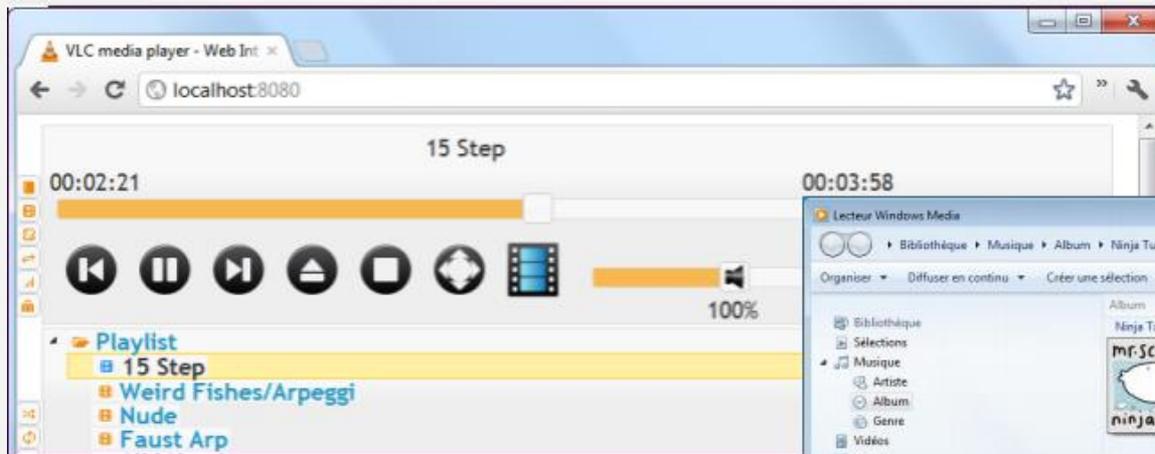
- La compatibilité est la capacité du logiciel à s'intégrer dans l'activité réelle des utilisateurs.
- Il faut qu'il y ait accord entre les caractéristiques des utilisateurs et des tâches, d'une part, et l'organisation des sorties, des entrées et du dialogue d'une application donnée, d'autre part.

## **Recommandations :**

- Parler le langage de l'utilisateur (éviter le jargon informatique)
- Utiliser des métaphores familières
- Agencer les éléments de l'interface en fonction de la tâche de l'utilisateur
- Présenter les informations de façon cohérente par rapport aux autres supports de travail (documents papier, formulaires, organigrammes, directives, etc.)
- L'accès aux fonctions doit être compatible avec la tâche de l'utilisateur

# 8. COMPATIBILITÉ

- A faire :



Activer Windows  
Accédez aux paramètres  
activer Windows.

# UNE CHECK-LIST (1)

Critère	Questions
<b>Guidage</b>	<ul style="list-style-type: none"><li>- L'utilisateur est-il assisté dans la façon de se servir du logiciel (en fournissant par exemple le format de saisie des données, une liste des valeurs possibles, etc.) ?</li><li>- Une aide en ligne est-elle proposée ?</li><li>- La documentation est-elle claire ?</li></ul>
<i>Incitation</i>	<ul style="list-style-type: none"><li>- L'utilisateur est-il amené à effectuer des actions spécifiques ?</li></ul>
<i>Groupement / Distinction</i>	<ul style="list-style-type: none"><li>- Les informations de même type sont-elles regroupées ?</li><li>- Distingue-t-on les données différentes ?</li></ul>
<i>Retour utilisateur</i>	<ul style="list-style-type: none"><li>- Le système fournit-il un retour aux actions de l'utilisateur ?</li><li>- Les opérations réalisées par le système sont-elles perceptibles ?</li></ul>
<i>Lisibilité</i>	<ul style="list-style-type: none"><li>- Les informations sont-elles correctement lisibles et interprétables ?</li></ul>

# UNE CHECK-LIST (2)

Critère	Questions
<b>Charge de travail</b> <i>-Concision</i> <i>-Actions minimales</i>	<ul style="list-style-type: none"><li>- Les activités de perception et de mémorisation sont-elles réduites au minimum ?</li><li>- Le nombre d'options dans les menus est-il raisonnable ?</li><li>- Les labels et textes sont-ils concis ?</li><li>- Les saisies sont-elles réduites au minimum ?</li><li>- Le nombre d'étapes pour atteindre un but est-il minimal ?</li><li>- Existe-t-il des raccourcis ?</li></ul>
<i>- Densité informationnelle</i>	<ul style="list-style-type: none"><li>- L'affichage demande-t-il un effort de perception ?</li><li>- Y a-t-il des informations inutiles à l'écran ?</li><li>- Les textes sont-ils rédigés de manière simple ?</li></ul>

# UNE CHECK-LIST (3)

Critère	Questions
<b>Contrôle explicite</b> - Actions explicites - Contrôle utilisateur	<ul style="list-style-type: none"><li>- Les fonctions sont-elles toujours explicitement activées par l'utilisateur ?</li><li>- Peut-il quitter, abandonner facilement ou interrompre un traitement en cours ?</li><li>- L'utilisateur peut-il revenir en arrière ?</li><li>- L'utilisateur maîtrise-t-il tous les traitements réalisés par le système?</li></ul>
<b>Adaptabilité</b> - Flexibilité	<ul style="list-style-type: none"><li>- Différents moyens sont-ils offerts à l'utilisateur pour déclencher les mêmes commandes ?</li><li>- Les commandes sont-elles également accessibles au clavier ?</li></ul>
-Expérience utilisateur	<ul style="list-style-type: none"><li>- L'utilisateur peut-il paramétrer le logiciel selon ses préférences ?</li><li>- Une alternative rapide est-elle proposée à l'utilisation des menus (par exemple des raccourcis clavier) ?</li></ul>

# UNE CHECK-LIST (4)

Critère	Questions
<b>Gestion des erreurs</b>	<ul style="list-style-type: none"><li>- Est-il possible d'explorer le logiciel sans risque ?</li><li>- L'impact des erreurs est-il minimisé ?</li></ul>
- <i>Protection</i>	<ul style="list-style-type: none"><li>- Le système offre-t-il des moyens de prévenir des erreurs (boutons grisés, liste des valeurs possibles, affichage des unités, ...) ?</li><li>- L'utilisateur est-il prévenu rapidement de son erreur ?</li><li>- L'utilisation du clavier est-elle minimale ?</li><li>- L'utilisateur est-il averti lors d'opérations dangereuses ?</li></ul>
- <i>Qualité des messages d'erreurs</i>	<ul style="list-style-type: none"><li>- Les messages sont-ils bien visibles ?</li><li>- La nature et les causes des erreurs sont-elles aisément identifiables ?</li><li>- Les moyens de correction sont-ils explicites ?</li></ul>
- <i>Correction</i>	<ul style="list-style-type: none"><li>- Les erreurs peuvent-elles être facilement corrigées ?</li><li>- Les éléments erronés sont-ils mis en évidence ?</li><li>- Existe-t-il un moyen de récupérer des données détruites ?</li></ul>

# UNE CHECK-LIST (5)

Critère	Questions
<b>Homogénéité Cohérence</b>	<ul style="list-style-type: none"><li>- L'agencement des fenêtres est-il semblable (tracé régulateur) ?</li><li>- Les couleurs, les icônes, les éléments graphiques et les polices de caractères sont-ils utilisés de façon cohérente ?</li><li>- Les formats de présentation des données sont-ils constants ?</li><li>- Un vocabulaire uniforme est-il utilisé dans l'ensemble des fenêtres ?</li><li>- Le fonctionnement de la souris est-il cohérent ?</li><li>- Le logiciel est-il cohérent du point de vue de son comportement ?</li></ul>

# UNE CHECK-LIST (6)

Critère	Questions
<b>Signifiante des codes et dénominations</b>	<ul style="list-style-type: none"><li>- Tous les éléments de l'interface sont-ils facilement interprétables par l'utilisateur ?</li><li>- L'application respecte-t-elle les standards en vigueur dans le domaine d'application ?</li><li>- Les abréviations et autres codages sont-ils compréhensibles ?</li></ul>
<b>Compatibilité</b>	<ul style="list-style-type: none"><li>- Le logiciel correspond-il au contexte d'utilisation ?</li><li>- Est-il adapté au profil des utilisateurs visés ?</li><li>- Le vocabulaire de l'interface est-il celui employé par les utilisateurs ?</li><li>- Les informations sont-elles présentées de manière cohérente par rapport aux autres supports de travail ?</li><li>- L'accès aux commandes est-il adapté au contexte de réalisation de la tâche ?</li></ul>