

CHAPITRE 1: Interface Homme - Machine et Notion d'interaction.

*** Terminologie utilisée:** La Terminologie concerne la Conception visuelle et fonctionnelle des interfaces interactives incluant l'interface graphique, le design d'application, la GUI (Interface Graphique Utilisateur), l'IHM (Interface Homme-Machine) et les chartes ergonomiques qui définissent les Règles pour concevoir des interfaces intuitives et ergonomiques.

*** Objets de l'IHM:**
- Les objets de l'interface sont présentés sous figures.
Communication facile et intuitive avec l'utilisateur.

*** Objectifs d'une IHM:**
- L'IHM vise à faciliter l'accomplissement des tâches de l'utilisateur de manière efficace, productive, sécurisée et agréable, tout en favorisant une courbe d'apprentissage rapide du système.

*** Qu'est-ce que l'IHM?**
- L'IHM est une application informatique interagissant avec les utilisateurs en intégrant leurs entrées pour générer une représentation perceptible de son état interne, différente des systèmes batch par leur nature interactive et transactionnelle.

* Historique de l'IHM:

1960: époque de la carte perforée.

1963: Travaux de **JVAN Sutherland** (Sketchpad a été le premier programme à proposer une interface graphique).

1968: A Stanford Research Institute: **Douglas Engelbart** (Premier du mode graphique comme interface).

- Invention de la Souris
- Travaux sur l'Hypertexte
- Invention du Concept de la fenêtre.

Souris + Fenêtre = Naissance de l'interface graphique.

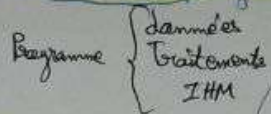
1970: XEROX (Conception langage qui manipule la fenêtre).

1978: Macintosh (Apple) bureau, icônes, technologie. WYSIWYG: what you see is what

you get.

1990: explosion grâce à windows.

* Structure d'un programme:



Type Interfaces:
- orienté Texte (Terminal)
- orienté fenêtres (windows)
- orienté document (Media/WEB)

* Types de Programmes:

* 3 Types:

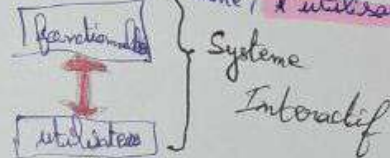
- Les programmes qui ne communiquent pas avec les utilisateurs. (Thermomètre, baromètre)
- Les programmes qui communiquent de façon indirecte. (Imprimante)
- Les programmes Interactifs. (Affichage du résultat sur l'écran, Saisie de données par l'utilisateur par l'entrée d'une commande par l'utilisateur).

* Interaction Homme - Machine:

Ensemble des Actions nécessaires pour assurer la communication entre le système dit interactif et son utilisateur.

* Interface VS Interaction:

- L'interface désigne le vecteur de communication (le moyen, l'outil, dispositif technique ou média).
- L'interaction met en jeu deux éléments du vecteur de communication: fonctionnalités proposées par la machine, l'utilisateur.



* Importance d'un système interactif:

- L'interaction d'un système va bien au-delà des éléments visuels: elle repose sur la fluidité des actions, la rapidité, l'adaptabilité de l'utilisateur, le taux d'erreur et la satisfaction subjective pour répondre aux besoins de l'utilisateur.

* Catégories de systèmes interactifs:

- Les styles d'interactions définissent les règles et les méthodes de communication propres à chaque système interactif, permettant de catégoriser leur mode d'interaction spécifique.

- ya 03 Styles d'interaction:
 - Style Conversationnel. (Terminal, ...)

- Style par manipulation directe (Menu, Fourniture)
- Style par reconnaissance de traces (WYSIWYG, édition de document, Interaction iconique, WIMP, WinSaaS, Icons)
- Style par reconnaissance de traces (Interface à Style, Tactile)
- + Autres Catégories :
- Intéact
- Interaction Sonore
- " Temporelle
- " See Through
- Réalité virtuelle.
- Interface 3D.
- Interaction 3D.
- Réalité augmentée.
- Interaction collaborative.
- " Multimédia.
- Crossing and steering.

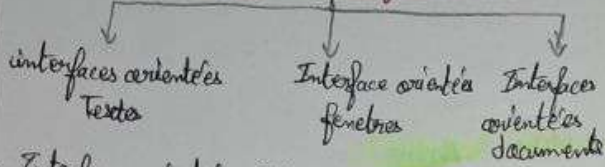
FIN

CHAPITRE 2: Conception d'interfaces Homme-Machine

* Systèmes Interactifs:

- Les Systèmes interactifs informatiques s'appuient sur les interactions utilisateur pour générer une représentation interne perceptible, dépassant l'engorgement pour s'adapter activement à l'utilisateur.
- Le Système interactif est composé de 2 parties:
 - * L'interface utilisateur
 - * Le Noyau fonctionnel.

* Styles d'interfaces



- Interface orientée fenêtres: Concept WIMP: window, icône, Menu, painteur de souris.



* 4 Types de fenêtres:

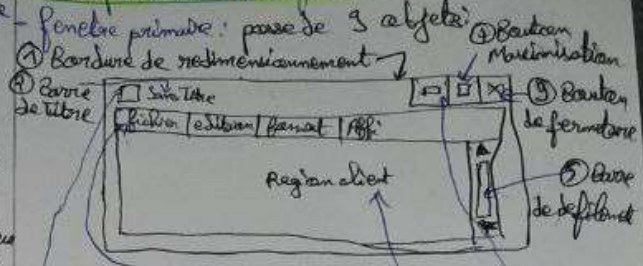
fenêtre primaire: c'est elle qui génère toutes les autres fenêtres.

fenêtre Secondaire: c'est une interface graphique distincte qui s'ouvre au-dessus ou à côté de la fenêtre principale d'une Application, offrant des fonctionnalités complémentaires ou des actions spécifiques.

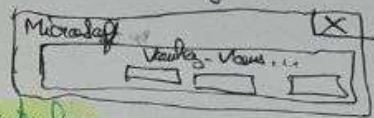
* **Boîtes de dialogue:** une fenêtre à travers laquelle l'application communique ou établit un dialogue avec l'utilisateur. (Modale: Tu ne peux pas la fermer sans l'accepter. Tu peux la fermer)

* **Boîte de message:** une fenêtre qui permet d'afficher un conseil, une information, un avertissement.

* Composition de ces fenêtres:



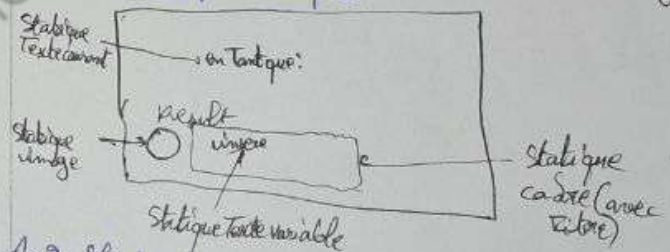
- Boîte de dialogue: possède 5 objets: une barre de titre, une Région client, Menu Système, Barre Menu, Bouton de minimisation.



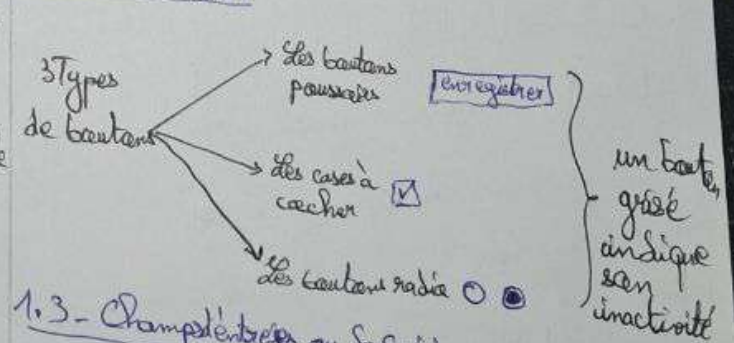
* Les Contrôles:

- Deux catégories de Contrôles:
 - 1. Contrôles de base: utilisés dans la gestion des données de l'application de gestion.
 - 2. Contrôles Supplémentaires: utilisés pour améliorer la lisibilité d'une boîte de dialogue ou d'un formulaire.

1.1 - Les champs statiques:

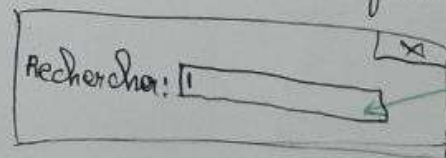


1.2 - Les boutons:



1.3 - Champ d'entrée ou de saisie:

Zone permettant de saisir des informations.



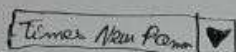
1.4- Les list box ou boîtes à listes:

il existe 4 types de list box:

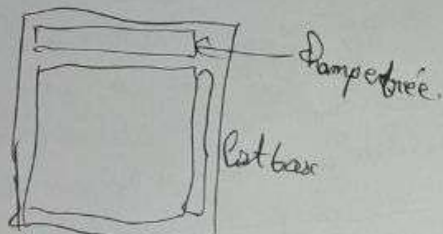
+ List box simple: sa totalité apparaît, elle peut être à sélection unique ou multiple.



+ List box déroulante: permet de cacher à l'utilisateur la liste des éléments. Seuls s'affiche que le dernier et élément sélectionné, elle peut être en position fermée ou ouverte.



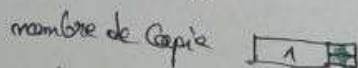
+ Combobox Simple: une superposition d'un champ d'entrée et d'une list box.



+ Combobox déroulante: une combinaison d'un champ d'entrée et une list box déroulante.



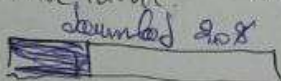
2.1- **Cont Selecteur radio:** il permet à l'utilisateur de saisir une seule valeur.



2.2- **Courseur gradué:** permet d'indiquer une valeur précise.



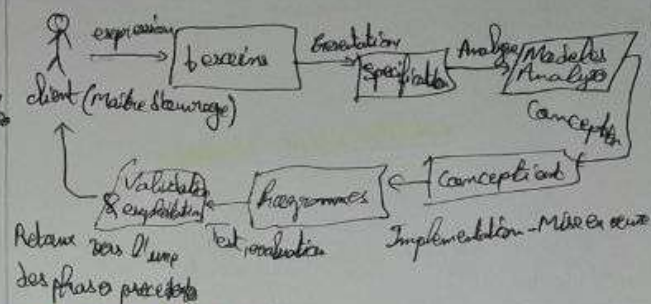
2.3- **La jauge:** elle permet d'afficher l'état d'avancement d'une tâche.



+ Cycle de vie d'une Application:

Le cycle de vie d'une Application désigne la période comprise entre le moment de commande de l'application et le moment de son abandon par l'utilisateur.

Les différentes phases du cycle de vie d'une application:



+ Methodes Agiles:

Les Methodes agiles mettent l'accent sur l'aspect humain et interactif, privilégiant la livraison rapide de produits opérationnels au lieu d'une documentation excessive, favorisant la collaboration continue avec le client plutôt qu'une simple négociation de contrat et privilégiant la réactivité face au changement par rapport à un suivi strict de processus, accueillant ainsi les évolutions nécessaires pour répondre aux demandes du client.

+ Methode centrée utilisateur:

une approche de conception qui place les besoins, préférences et comportements des utilisateurs au cœur du processus de développement. Elle vise à comprendre profondément les utilisateurs finaux pour créer des produits.

+ Modeles d'utilisateurs:

utilité: si elle répond aux besoins de l'utilisateur.

utilisabilité: c'est la qualité et la facilité d'utiliser l'interface.

Le Modèle d'utilisateur en IHM est un outil conceptuel qui aide les concepteurs à mieux comprendre les utilisateurs finaux.

à anticiper leurs besoins et à créer des interfaces plus conviviales et adaptées.

* Les Aspects du modèles utilisateurs:

- Les Modèles utilisateurs décrivent le profil des utilisateurs cibles, les scénarios d'utilisation, les personas représentant des groupes d'utilisateurs spécifiques, les adaptations possibles de l'interface en fonction des préférences individuelles, et ils servent aussi à tester et itérer les interactions avec l'interface avant sa mise en œuvre complète.

* Processeur Humain:

- Le Modèle processeur Humain basé sur la psychologie cognitive, évalue l'utilisabilité en représentant l'être humain comme un système de traitement d'informations régi par des Règles. Il comporte 3 sous-systèmes indépendants (sensoriel, moteur et cognitif), chacun équipé d'un processeur et d'une mémoire.

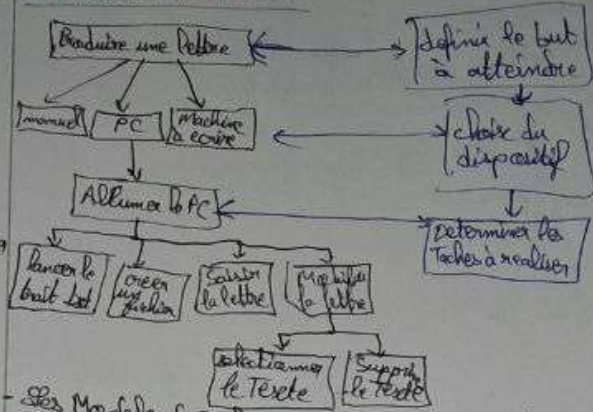
* Principales Mémoires Humaines:

- **Mémoire à court terme:** est une capacité limitée à retenir une petite quantité d'informations pour une durée limitée.
- **Mémoire à long terme:** stocke des informations de manière quasi-limitée sur une durée beaucoup plus longue, de la connaissance générale aux souvenirs personnels, avec une capacité de rétention pratiquement illimitée.

* **Système Cognitif:** peut être considéré comme un système de traitement des Informations, elles passent par 3 stades successifs:

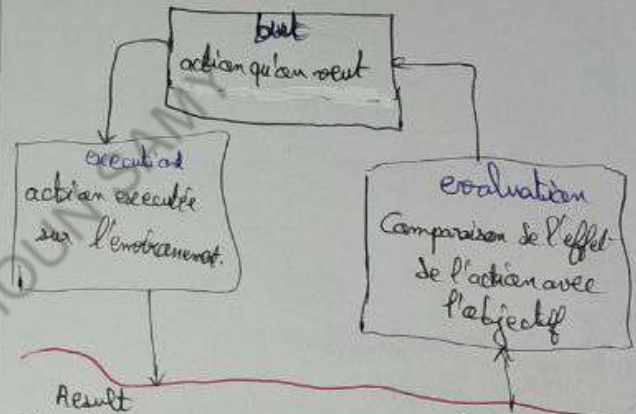
- Mémoire sensorielle.
- Mémoire à court terme.
- Mémoire à long terme.

* Modèles de Tâches:



- Les Modèles de tâches sont des structures arborescentes dans les nœuds sont des buts et les sous-arbres sont des procédures pour atteindre le but.

* Action de Norman:



- Donald Norman émet 7 étapes pour l'accomplissement d'une tâche:

- 1- établissement du but
- 2- formation d'une intention
- 3- spécification d'une suite d'actions
- 4- exécution des actions
- 5- Perception de l'état du Système
- 6- Interprétation de l'état du Système
- 7- Évaluation de l'état par rapport au but fixé.

* **Notion de gouffre:** Distance d'exécution et distance d'évaluation.

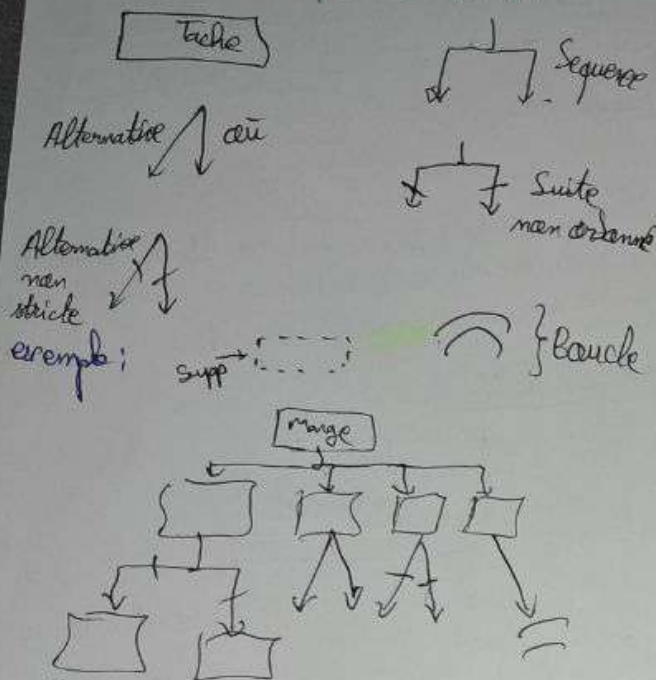
- le gouffre d'exécution représente la distance entre les actions que l'utilisateur doit effectuer et la manière dont il les exécute réellement.

Les différentes étapes sont réparties en 2 phases:

- phase d'exécution
- phase d'évaluation

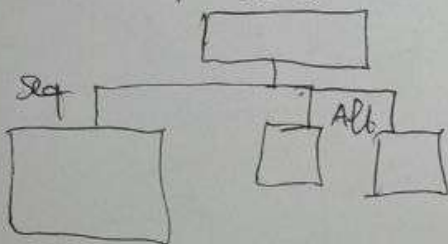
tandis que le graphique d'évaluation désigne la distance entre les informations fournies par le système et la compréhension de ces informations par l'utilisateur.

* Méthode de Scapin-MAD et HTA:



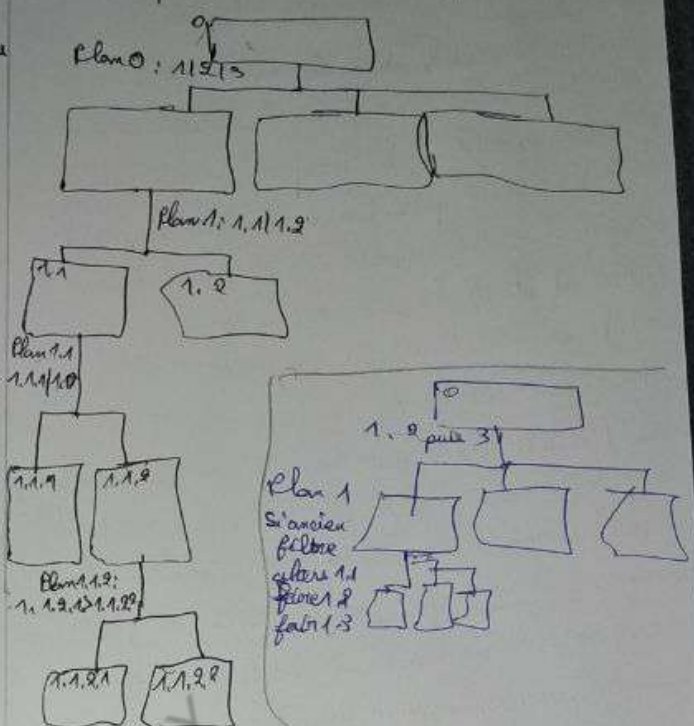
* Méthode MAD:

SEQ: Tâche Séquentielle
 PAR: Tâche parallèle
 BOUCLE: Tâche itérative
 ALT: Tâche alternative
 FAC: Tâche facultative
 SIM: Tâches Simultanées.
 PRIOR: Tâche prioritaire



* Méthode HTA:

Plan: chaque branche est un plan.



* Arbre des Tâches:

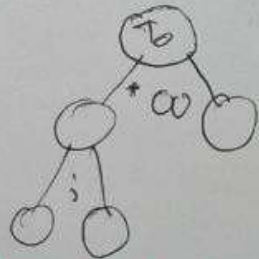
- Tâche demandée et indiquée
- Tâche demandée mais non indiquée
- Tâche non demandée et non indiquée

* Les actions:

mo: mise en oeuvre.
 a: acquisition
 c: Contrôle
 t: traitement

ET, OU, *, opt, ;, <>

T₁ puis T₂ au même temps



* IHM abstraite: définit la structure spatiale du système traduit le modèle de tâches en éléments d'interface, et intègre la gestion spatiale.

FIN

CHAPITRE 4: Les Règles ergonomique dans les IHM

* Qu'est ce que c'est l'ergonomie?

- l'ergonomie IHM, aussi appelée 'utilisabilité', est une science visant à rendre les fonctions d'un système facilement maniable et utilisable. Elle repose sur l'étude du comportement des utilisateurs pour concevoir des systèmes interactifs répondant à leurs besoins, favorisant leur bien-être physique et psychologique lors de l'utilisation de l'app.

* Objectif de l'ergonomie:

- confort, santé, efficacité.

* Interet(s) de l'ergonomie:

ergonomie physique: fonctionnement et comportement

ergonomie cognitive: étude de comportement de l'homme avec la machine.

ergonomie logicielle: se situe à côté de l'ergonomie cognitive: l'application doit être utile et utilisable (performance, satisfaction, facilité)

* Niveaux de l'ergonomie logicielle:

Niveau 1: ergonomie de surface: partie visible.

Niveau 2: ergonomie de profondeur: partie cachée

* Principe d'ergonomie:

- **Cohérence**: même concept de façon similaire.
- **Concision**: limiter l'intervention de l'utilisateur et de limiter les erreurs.
- **retour d'informations**: Toute action doit mener à un retour.
- **structuration des Activités**: Toujours proposer des fonctions simples.
- **flexibilité**: Adapté à chaque utilisateur.
- **gestion des erreurs**: orienter l'utilisateur de sorte à pouvoir résoudre les problèmes

* Composants importants pour les fenêtres:

1. Titre

2. Barre d'actions (Barre de menus)

3. Région client

4. Barre d'utils.

* Règles de compatibilité pour les fenêtres: (liste...)

* Sa Barre d'action: Menu principal.

1. Taille de la barre d'action: elle doit occuper une seule ligne dans la fenêtre.

2. Taille menus: la largeur des menus se limite à une dizaine d'options, la profondeur des menus doit pas dépasser 3 niveaux.

* Région client:

3. existence de la région client: si elle n'est pas utilisée, elle doit être supprimée de la fenêtre primaire.

* Boîte de dialogue:

4. Nombre de Contrôles: max 30 contrôles actifs.

5. Taille et position des boîtes de dialogue: elle doit occuper un minimum sur la surface du desktop, elle ne peut pas être redimensionner.

6. Position et regroupement des Contrôles: disposés par famille, alignés verticalement à gauche et à droite et chaque contrôle a une légende (titre) - Combobox et list box en haut.

- case à cocher et bouton radio alignés.

7. Mise en Majuscule: Première lettre en Maj pour le reste minuscule, si abréviation tout en Maj

8. Valeurs initiales: donner une valeur initiale pour chaque contrôle.

9. Types de list box: utiliser les list box et combobox de préférence aux listes déroulantes.

- les listes simples doivent permettre au moins 3 éléments sans scrolling.

10. couleur et disposition des boutons: ils doivent être placés dans le même endroit et ordre.

- Pour l'action par défaut de la boîte de dialogue (Annulation en cas de ...)

11- Taille et structuration des champs de Saisie:
la Taille dépend du type et la longueur du champ des données.

12- couleurs et polices: principe de cohérence.

13- aide en ligne: info - bulle, ...

très visible et accessible	très visible et accessible
très visible et accessible	très visible et accessible
très visible et accessible	très visible et accessible

Titre	
Commandes	
Zone de Travail	Zone de Messages

Quit

* Les Heuristiques de Nielsen:

- 1- visibilité du statut système: un indicateur clair montrant qu'un fichier est en cours de téléchargement.
- 2- Correspondance avec le monde réel: utiliser des icônes ressemblant aux objets du monde réel (icône pour élement supprimer).
- 3- Contrôle utilisateur et liberté: offrir une option "Annuler" après avoir initié une action pour permettre aux utilisateurs de revenir en arrière.
- 4- cohérence avec les standards: utiliser des conventions communes pour les boutons (exemple: une flèche pointant vers le bas pour indiquer un menu déroulant).
- 5- Prévention des fautes pas et erreurs: demander une confirmation avant de supprimer des éléments.
- 6- Reconnaissance plutôt que de se souvenir: Afficher les options de Menu au lieu d'attendre que l'utilisateur se souvienne des Commandes spécifiques.
- 7- flexibilité et facilité d'usage: offrir différentes méthodes pour effectuer une même tâche, par exemple via un raccourci clavier ou un bouton dans le menu.
- 8- Esthétique et minimalisme: Éliminer les éléments d'interface inutiles pour simplifier la vue de la fenêtre principale.
- 9- Aide à la reconnaissance, au diagnostic et à la réparation des erreurs: fournir des messages d'erreurs

explicites et des suggestions pour résoudre les problèmes rencontrés par l'utilisateur.

* Évaluation des IHM: (Dernier point Après les Règles de Consistance)

1. Critères de l'utilisabilité:

- efficacité
- efficacité (entre min une ressource)
- Satisfaction de l'utilisateur
- facilité d'apprentissage
- facilité d'utilisation
- fiabilité

2- Évaluation analytique:

- jugements d'experts.
- utilisation de référents.
- évaluation heuristique.

3- Évaluation expérimentale:

- observation et Recueil de données.
- entretiens et questionnaires.
- 1- visibilité des couleurs.
- 2- Affichage du Texte
- 3- Répartition Selon la lecture de l'écran.
- 4- charte graphique: pour assurer la cohérence (couleur, image, ...)
- 5- Répartition des éléments: les éléments importants puis les éléments fréquemment utilisés, les autres éléments à la fin.

4- Étude ergonomique:

Barre de menus:

Commandes importantes	Commandes fréquemment utilisées	Commandes annexes
-----------------------	---------------------------------	-------------------

Palette d'outils:

outils importants
outils fréquemment utilisés
outils annexes

Sens de lecture

- Analyse de la Tâche et des Commandes d'utilisateurs auxquels est destiné le système.
- Étude des besoins utilisateurs
- ② - Détermination de la nature de utilisateurs

- Conception: Respect des principes ergonomiques:
- Conception se fait indépendamment de la Technologie

Suite Etude:

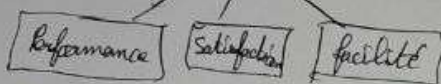
1- La Navigation:

Quelques pages pour obtenir une utilisabilité maximale d'une Application:

- l'efficacité, la facilité avec laquelle l'utilisateur atteint son objectif.
- La Satisfaction ressentie par l'utilisateur.
- la facilité d'apprentissage.
- la fiabilité.

2- Principes pratiques ergonomiques:

l'utilité et l'utilisabilité



3- Utilisation des couleurs:

utiliser 8 couleurs différentes au max (4 mieux)

4- Parcours Visuel Image: (Ex @ chapitre 4) Après 13s



5- Interaction HM:

Loi de Hick: Plus l'utilisateur a des choix plus il met plus de temps pour prendre une décision.

Loi de Fitts: le temps nécessaire pour atteindre une cible dépend à la fois de sa distance et de sa taille.

6- Alignement:

- libellé à gauche du champ
- " au dessus du champ
- " fusionné avec le champ

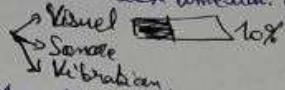
* Règles de Bartson et Shapiro:

1- Guidage:

1.1- incitation: utiliser des éléments visuels pour encourager l'utilisateur à effectuer une action spécifique. (griser les commandes non disponibles, afficher l'acte (ij mm/aaaa))

1.2- groupement / distinction: organiser visuellement les éléments similaires tout en les distinguant pour faciliter la compréhension (format, localisation)

1.3- Feedback immédiat: fournir une réponse immédiate



1.4- visibilité: assurer que le contenu est clairement visible et compréhensible.

2- charge de travail: optimiser le nbr d'échans

2.1- Breveté: utiliser des Menus et des actions concises et directs.

2.1.1- Concision: largeur (cm)

2.1.2- Actions Minimales: ne pas demander une information qui peut être déduite par le logiciel.

2.2- Densité informationnelle: éviter de surcharger l'interface avec trop d'informations à la fois.

3- Contrôle explicite: Permettre à l'utilisateur d'avoir un contrôle direct et clair avec le système

4- Adaptabilité: offrir des options pour s'adapter aux préférences et aux besoins de l'utilisateur.

5- Gestion des erreurs: Fournir des messages d'erreur clairs et des moyens pour corriger les erreurs.

6- Homogénéité / cohérence: Maintenir une uniformité dans la conception pour que les éléments se comportent de manière non attendue.

7- Signification des codes et dénomination: utiliser des termes et des symboles familiers et significatifs pour les utilisateurs.

8- Compatibilité: Assurer que l'interface est compatible avec les connaissances et les compétences des utilisateurs ainsi qu'avec d'autres systèmes qu'ils utilisent.

* Les Règles d'or de Cantory :

- 1- Lutter pour la cohérence : Maintenir une uniformité dans la conception de l'interface pour que les éléments se comportent de manière prévisible et attendue par l'utilisateur.
- 2- Lutter pour la concision : utiliser des messages et des actions courts et directs pour éviter tout excès d'informations ou de complexité.
- 3- Réduire la charge cognitive : simplifier l'interface et les actions pour réduire la charge mentale de l'utilisateur pendant l'utilisation.
- 4- Mettre le contrôle entre les mains de l'utilisateur : offrir des fonctionnalités permettant à l'utilisateur de contrôler et d'influencer le système selon ses besoins.
- 5- Souplesse d'utilisation : offrir différentes options et voies pour atteindre un objectif afin de s'adapter aux différentes préférences des utilisateurs.
- 6- Structurer l'interaction : organiser l'interaction entre l'utilisateur et le système de manière claire et logique pour faciliter la compréhension et la progression des tâches.
- 7- Prédire les erreurs : Concevoir l'interface de manière à anticiper et prévenir les erreurs potentielles de l'utilisateur en fournissant des indications et des guides appropriés.

[FIN]