Introduction aux IHM

De la ligne de commande aux interfaces conversationnelles

Séquence 1 – Cours (1h15)

BUT2 Informatique – Ressource IHM

Année 2026–2027

IUT de Fontainebleau

Objectifs et déroulé de la séquence

À la fin de cette séquence, vous saurez :

- distinguer UI et UX et chiffrer le coût d'une mauvaise interface ;
- énoncer et appliquer les lois de Hick, Fitts et Jakob ;
- situer les grands paradigmes d'interaction (CLI, WIMP, Web, tactile, conversationnel) ;
- reconnaître les grands courants visuels (skeuomorphisme, flat, ...).

Séance	Contenu
Cours 1h15	Principes, lois, histoire
TD 1h15	Calculs Fitts/Hick, audit d'un site
TP 1h15	CLI vs GUI, Flat vs Skeuo

Séquence 2 : **Création de wireframes** (on réutilisera tout ce qu'on voit ici).

Plan du cours

L'UI, ça compte !

Principes et lois de l'UX

Une brève histoire des interfaces

Les grandes tendances visuelles

La nouvelle frontière : l'IA

Synthèse

L'UI, ça compte !

Pourquoi ce cours ? (vous n'êtes pas designers. . .)

- L'interface est le seul point de contact entre votre code et l'utilisateur.
- Une bonne IHM est un moteur d'**adoption**, d'**efficacité** et de **satisfaction**.
- Une mauvaise UI a aussi un coût technique pour vous, développeur · ses :
  - **code plus complexe** : contournements (*workarounds*) pour pallier la conception ;
  - **maintenance difficile** : un parcours illogique donne une machine à états illisible ;
  - **faux bugs** : des tickets « bug » qui sont en réalité des problèmes d'utilisabilité.

À retenir

Vous n'êtes pas designers, mais des développeur · ses **conscient · es de l'impact** de leur travail sur l'usager final.

UI ou UX ?

UI – User Interface : le « comment »

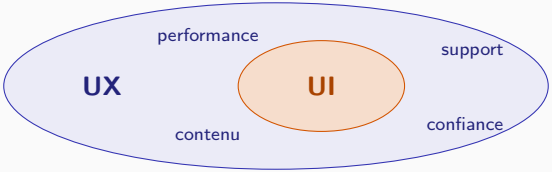
La partie **visible et tangible** avec laquelle on interagit.

Boutons, menus, formulaires, agencement, typographie, couleurs.

UX – User Experience : le « ressenti »

L'expérience **globale**, pratique et émotionnelle, avec le produit.

Est-ce facile de s'inscrire ? Est-ce rapide ? Ai-je confiance pour saisir ma carte bancaire ?



Une bonne UI *contribue* à une bonne UX, mais ne suffit pas.

Le coût d'une mauvaise UI

- **Frustration** → abandon, mauvais avis sur les stores.
- **Erreurs** → perte de données, mauvaise commande.
- **Formation** plus longue → coût pour les logiciels métier.
- **Charge cognitive** → on réfléchit à l'outil au lieu de la tâche.
- **Support** surchargé → appels que l'interface aurait dû éviter.
- **Image de marque** dégradée.

? Question à la salle

Citez une interface (site, appli, borne, machine à café…) qui vous a fait perdre du temps cette semaine. *Pourquoi ?*

Principes et lois de l'UX

Quatre principes qui font une bonne UI

Hiérarchie visuelle

Ce qui est important est plus gros, plus contrasté, placé en premier.

Affordance

L'élément suggère sa fonction : un bouton « ressemble » à un bouton, un lien est souligné.

Feedback

Chaque action produit une réaction immédiate : état *pressed*, barre de chargement, message de succès.

Cohérence

Mêmes icônes, mêmes actions, mêmes emplacements dans toute l'application.

L'anti-principe : les *dark patterns*

Des designs **trompeurs** qui poussent à une action non souhaitée : « Accepter » énorme et coloré, « Refuser » minuscule et gris ; abonnement facile, résiliation introuvable.

Loi de Hick (Hick & Hyman, 1952)

Énoncé

Le temps de décision **augmente avec le nombre** de choix (de façon logarithmique).

$$T = a + b \cdot \log_2(n + 1)$$

n : nombre de choix équiprobables, a, b : constantes mesurées.

En pratique :

- limiter les options visibles, **regrouper** en catégories ;
- mettre en avant **un** choix recommandé ;
- « Continuer avec Google » + « avec un e-mail » plutôt que 15 fournisseurs.

8 entrées à plat



2 menus × 4



*Attention : la loi vaut pour des choix **connus**. Pour un menu inconnu, l'utilisateur **balaie** la liste (temps $\propto n$) : le regroupement aide encore plus. (on calcule en TD)*

Loi de Fitts (Paul Fitts, 1954)

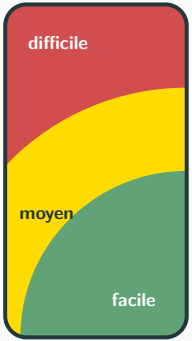
Énoncé

Le temps pour atteindre une cible dépend de sa **distance** D et de sa **taille** W :

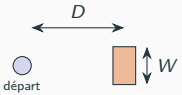
$$T = a + b \cdot \underbrace{\log_2(D/W + 1)}_{\text{indice de difficulté}}$$

En pratique :

- actions principales **grandes** et **proches** ;
- « Acheter » plus gros que « Annuler » ; « Supprimer » loin de « Enregistrer » ;
- **bords et coins** d'écran : cibles « infinies » à la souris ;
- mobile : actions clés **en bas**, sous le pouce.



Zone du pouce
(droitier, une main)



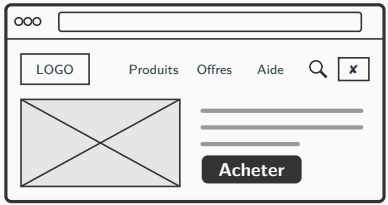
Loi de Jakob (Jakob Nielsen, 2000)

Énoncé

Les utilisateurs passent l'essentiel de leur temps **sur les autres sites**. Ils préfèrent donc que le vôtre fonctionne **comme ceux qu'ils connaissent déjà**.

Conventions à respecter :

- logo en haut à gauche, cliquable → accueil ;
- panier / compte en haut à droite ;
- loupe = recherche, ✖ = fermer, lien = souligné ou coloré ;
- menu « burger » sur mobile.



À retenir

Innover, oui – mais l'innovation doit servir l'utilisateur, pas votre ego.

Quiz express (2 min)

Quelle loi ou quel principe est en jeu ?

1. Le bouton « Valider la commande » fait 12 px de haut sur mobile.
2. Le menu d'un site d'université propose 47 liens au même niveau.
3. Sur un site e-commerce, le panier est en bas à gauche de la page.
4. Après clic sur « Envoyer », rien ne change pendant 4 secondes.
5. « Se désabonner » est un lien gris clair de 9 px en pied de page.
6. Un texte en bleu non cliquable au milieu des liens bleus.

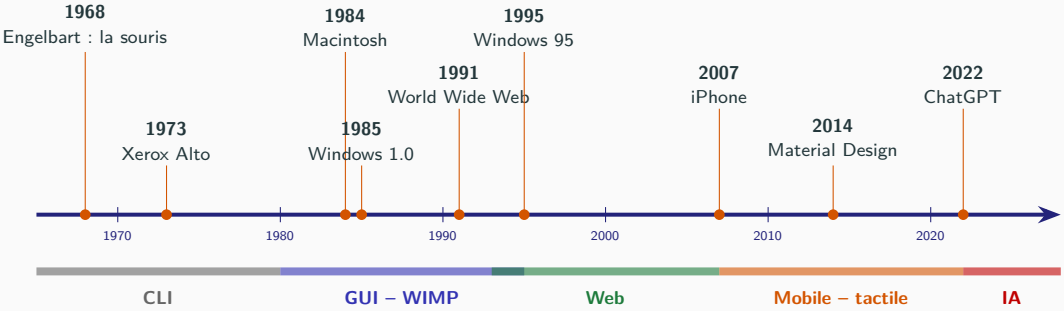
Quiz express (2 min)

Quelle loi ou quel principe est en jeu ?

1. Le bouton « Valider la commande » fait 12 px de haut sur mobile. — Fitts
2. Le menu d'un site d'université propose 47 liens au même niveau. — Hick
3. Sur un site e-commerce, le panier est en bas à gauche de la page. — Jakob
4. Après clic sur « Envoyer », rien ne change pendant 4 secondes. — Feedback
5. « Se désabonner » est un lien gris clair de 9 px en pied de page. — Dark pattern (+ Fitts)
6. Un texte en bleu non cliquable au milieu des liens bleus. — Affordance / cohérence

Une brève histoire des interfaces

Vue d'ensemble



Avant le graphique : l'ère de la ligne de commande (CLI)

- Interaction **purement textuelle** : il faut connaître la syntaxe exacte.
- Née quand CPU, mémoire et affichage étaient **rares et chers**.
- + puissance, précision, **scriptabilité**, automatisation
  → toujours omniprésente chez les devs : `git`, `docker`, `ssh`, `npm`...
- − courbe d'apprentissage raide, **aucune découvrabilité**.

À retenir

Le monde appartenait à ceux qui connaissaient le manuel par cœur.

```
$ ls -la
drwxr-xr-x 4 pv staff 128 .
-rw-r-r- 1 pv staff 2.1K notes.md
$ grep "error" app.log | wc -l
42
$ git commit -m "fix: login"
$
```

Puissant pour l'expert, **opaque** pour le novice : aucun indice visuel sur ce qui est possible.

La révolution graphique : de Engelbart au Xerox Alto

- **1968** – Douglas Engelbart (SRI) présente la *« Mother of all demos »* : souris, fenêtres, hypertexte, visioconférence.
- **1973** – Le **Xerox Alto** (PARC) : premier ordinateur pensé autour d'une interface graphique.
  - écran en **portrait** (format page A4), éditeur **WYSIWYG** ;
  - la **métaphore du bureau** : fichiers, dossiers, corbeille ;
  - au PARC naissent aussi l'impression laser, Ethernet, Smalltalk (la POO).

Le basculement cognitif clé

On passe du rappel (se souvenir d'une commande) à la reconnaissance (reconnaître une icône, un menu). Reconnaître est beaucoup moins coûteux que se souvenir.

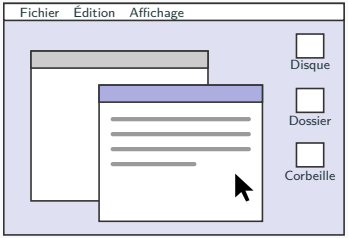
Le paradigme WIMP

Windows plusieurs contextes visibles en même temps : le multitâche visuel.

Icons raccourcis graphiques vers fichiers, dossiers, applications.

Menus les fonctions deviennent **découvrables** sans mémoriser.

Pointer manipulation **directe** : cliquer, glisser-déposer.



Ce paradigme domine le *desktop* depuis 40 ans. C'est encore celui de vos IDE.

1984–1995 : le GUI devient grand public

1984 – Macintosh

Apple n'invente pas : elle **simplifie et intègre**.

Souris à **1 bouton** (3 sur l'Alto), barre de menus en haut d'écran, icônes de Susan Kare.

1985 – Windows 1.0

Une **surcouche graphique** à MS-DOS.

Fenêtres **en tuiles** (pas de superposition avant Windows 2.0). Moins fluide, mais début d'une domination.

1995 – Windows 95

Le point de bascule : des millions de foyers.

Menu Démarrer, **barre des tâches**, standardisation massive des conventions.

1987 – Naissance des *guidelines*

Apple publie les **Human Interface Guidelines** (cohérence, manipulation directe, feedback, métaphores), Microsoft suit dans les années 90. Objectif : que toute application tierce **se comporte comme une appli native**. On apprend l'OS **une fois**, puis on réutilise ce savoir partout ($\rightarrow$ loi de Jakob avant l'heure).

Aujourd'hui : *Apple HIG*, *Material Design* (Google), *Fluent* (Microsoft).

Les années 90 : l'autre révolution, le Web

1989–1991 : Tim Berners-Lee (CERN)

- but : partager des documents entre chercheurs ;
- primitive fondamentale : l'hyperlien → navigation **associative**, plus hiérarchique ;
- l'interface *est* le document.

1994–1995 : la guerre des navigateurs

- Netscape Navigator puis Internet Explorer ;
- balise `<img>` : le Web devient visuel.

Le « chaos créatif »

- mise en page détournée avec des `<table>` (pas encore de CSS utilisable) → cauchemar sémantique et accessibilité ;
- GIF clignotants, compteurs de visites, fonds en mosaïque, « en construction » ;
- boutons en relief (skeuomorphisme) pour rassurer.

Fossile parfaitement conservé : le site de *Space Jam* (1996). *Vous le recréerez en bonus…*

Le tournant mobile : de Windows Mobile à l'iPhone

Années 2000 : le desktop miniaturisé

BlackBerry (clavier physique), Nokia/Symbian (leader), Windows Mobile.

Bouton « Démarrer », minuscules croix de fermeture, menus contextuels...
Utilisation au **stylet** ou au trackball : pas pensé pour le doigt.

2007 : l'iPhone et le multi-touch

Fin de la manipulation **indirecte** : on touche le contenu.

Nouveau vocabulaire : *tap*, *swipe*, *pinch-to-zoom*, *long press*.
Grille d'icônes, une appli plein écran, pas de système de fichiers visible : **charge cognitive minimale**.

À retenir

Mobile-first : on ne « réduit » plus un logiciel desktop, on conçoit d'abord pour le mobile (contexte, écran, connectivité, **doigt de $\approx$ 1 cm**).

Android, Material Design et Responsive Design

Le défi de Google : une expérience cohérente sur des milliers de modèles (tailles, résolutions, fabricants).

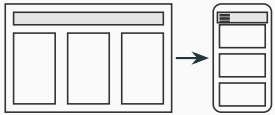
2014 – Material Design

Un **système de design** complet, pas un simple style.
Métaphore « papier et encre numériques » : surfaces avec **élévation** (ombres), mouvements cohérents, bouton d'action flottant (*FAB*) pour l'action principale.

2010 – Responsive Web Design (E. Marcotte)

Un **seul** site qui s'adapte à tous les écrans :

- **media queries** CSS ;
- **grilles fluides** (%, `flexbox`, `grid`) ;
- **images flexibles**.



Les grandes tendances visuelles

Du skeuomorphisme au verre

Skeuomorphisme
2000–2012



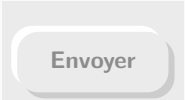
Flat design
2012 →



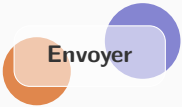
Material
2014 →



Neumorphism
2019–2020



Glassmorphism
2020 →



Skeuo : imiter le réel (étagère en bois d'iBooks, cuir de l'agenda) pour **rassurer** face au numérique.

Flat : l'interface assume sa nature numérique (iOS 7 en 2013, Metro de Windows 8). Le **contenu** d'abord.
Risque : perte d'affordance.

Neumorphism : éphémère car **contraste insuffisant** (accessibilité). **Glass** : flou et transparence pour la hiérarchie (macOS Big Sur, Windows 11, *Liquid Glass* d'Apple en 2025).

La nouvelle frontière : l'IA

De la manipulation directe à l'intention

Manipulation directe

Je clique « Fichier » → « Enregistrer sous. . .» → je nomme le fichier → « OK ».

Interaction par intention

« Sauvegarde ce document dans mon dossier projet sous le nom *rapport final*. »

Trois familles d'interfaces augmentées par l'IA :

1. **Conversationnelles** (CUI) : vocales (Siri, Alexa) ou textuelles (chatbots, assistants de code dans l'IDE).
2. **Génératives** : du composant UI généré à partir d'un *prompt* (ex. `v0.dev`), images générées.
3. **Proactives** : suggestions de réponses (*Smart Reply*), playlists personnalisées : l'UI **anticipe**.

Nouveaux enjeux pour les développeur · ses

- **Le prompt comme nouvelle API** : décrire précisément une intention devient une compétence.
- **Gérer l'incertitude** : l'IA se trompe, hallucine, n'est pas déterministe.
  → « Je ne suis pas sûr d'avoir compris, voici 3 options. . . »
- **Éthique et transparence** : dire clairement qu'on parle à une IA, expliquer ses recommandations.
- **Orchestrer plutôt que construire** : on assemble des services qui génèrent une partie de l'interface.

? Question à la salle

L'UI va-t-elle disparaître au profit d'une unique boîte de dialogue, ou devenir un « canevas intelligent » sur lequel l'IA et l'utilisateur collaborent ? *Pensez aux lois de Hick et Jakob. . .*

Synthèse

Synthèse de l'évolution

Ère	Interface	Métaphore	Interaction	Exemple
1960–70	CLI	Terminal	Texte, commandes	`bash`
1980–90	GUI	Bureau	Souris (WIMP)	Windows 95
1990–2000	Web	Page / document	Hyperlien, souris	Netscape
2007 $\rightarrow$	Mobile	Application	Tactile, gestes	iPhone OS
2020 $\rightarrow$	IA	Conversation	Langage naturel	Chatbots

À retenir

Chaque paradigme **s'ajoute** aux précédents sans les remplacer : vous utilisez encore une CLI tous les jours. Les lois de Hick, Fitts et Jakob s'appliquent à **tous**.

Pour la suite

TD – « L'inspecteur des lois »

- calculer avec Fitts et Hick ;
- reconnaître lois et principes sur des cas ;
- auditer `iut-fbleau.fr` ou `u-pec.fr` ;
- proposer une maquette papier corrective.

TP – « Duels d'interfaces »

- chronométrer GUI vs CLI sur un renommage en masse ;
- coder un formulaire de connexion en **flat** puis en **skeuomorphique**.

À apporter : ordinateur, compte sur la machine, chronomètre (téléphone).

Questions ?