Création de wireframes

Méthodologies et outils

Séquence 2 – 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 :

- expliquer ce qu'est (et n'est pas) un wireframe, et choisir un niveau de fidélité ;
- produire un wireframe **papier** lisible et **annoté** ;
- le formaliser dans un outil **numérique** (Figma ou Penpot) ;
- le traduire en **squelette HTML/CSS** sémantique ;
- choisir la bonne méthode selon le contexte.

Séance	Contenu
Cours 1h15	3 méthodologies
TD 1h15	Déconstruire, hiérarchiser
TP 1h15	« Ma Biblio » : papier → Figma → HTML/CSS

Fil rouge : les lois de Hick, Fitts et Jakob de la séquence 1 s'appliquent *dès* le wireframe.

Plan du cours

Introduction et concepts fondamentaux

Méthodologie 1 : le dessin (paper prototyping)

Méthodologie 2 : les outils numériques

Méthodologie 3 : le wireframe en code

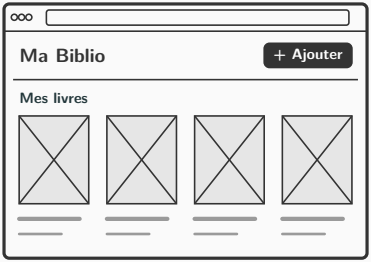
Synthèse et mise en pratique

Introduction et concepts fondamentaux

Qu'est-ce qu'un wireframe ?

Définition

Un **wireframe** (maquette *fil de fer*) est une représentation visuelle **schématique** de la structure d'une page web ou d'un écran d'application.



Il se concentre sur :

- la **structure** et la disposition des éléments ;
- la **hiérarchie** de l'information (qu'est-ce qui est important ?) ;
- la **fonctionnalité** (que peut faire l'utilisateur ?).

Important

Un wireframe **ignore délibérément** le style, les couleurs et les images, pour ne pas distraire de l'objectif : l'**architecture de l'information**.

Pourquoi faire des wireframes ?

Communication

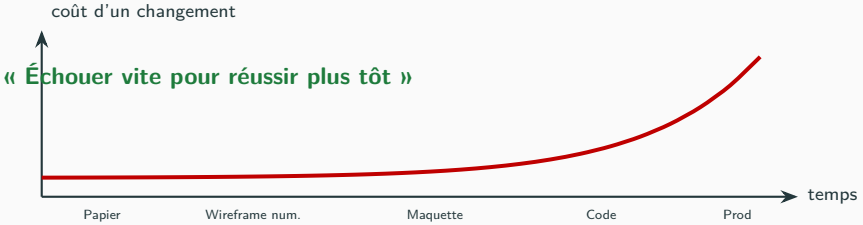
- aligner designers, devs et clients ;
- une base de discussion **concrète** plutôt qu'un cahier des charges abstrait.

Validation

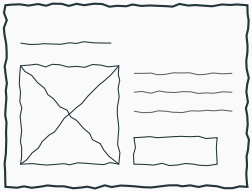
- tester les parcours **très tôt** ;
- itérer vite et à faible coût.

Efficacité

- économiser temps et argent ;
- servir de **plan** aux UI designers et aux devs.

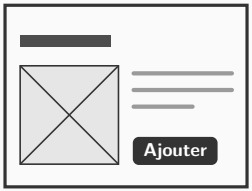


Niveaux de fidélité



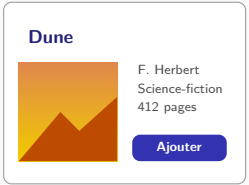
Low-Fidelity

Papier, tableau blanc. Rapide, jetable. Idéal pour le **brainstorming** et l'exploration.



Mid-Fidelity

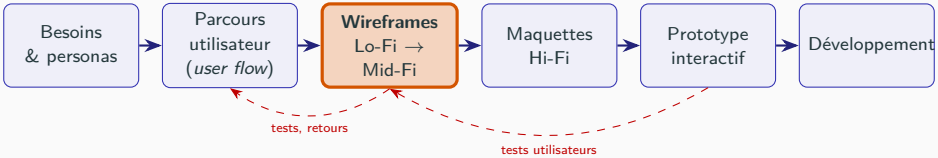
Outil numérique (Figma, Penpot, Balsamiq). **Niveaux de gris**, blocs, texte standard. Le plus courant pour la documentation.



High-Fidelity

Proche du produit final (souvent appelé **maquette** ou *mockup*) : couleurs, typo, vraies images. Base des prototypes interactifs.

Où se situe le wireframe dans un projet ?



À retenir

Le processus est **itératif** : on revient en arrière tant que c'est encore peu coûteux. Un wireframe sans **parcours** (d'où vient l'utilisateur ? où va-t-il après ?) n'a pas de sens.

Méthodologie 1 : le dessin (paper prototyping)

Le cerveau, le papier, le crayon

La méthode la plus fondamentale

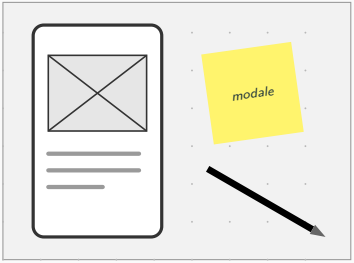
La plus rapide, et souvent la plus efficace, pour passer d'une idée abstraite à un concept tangible.

« Il s'agit de démarrer une conversation, pas de créer un chef-d'œuvre. »

L'objectif n'est pas la beauté, mais la **clarté**, la **structure** et l'**itération rapide**.

Le matériel essentiel

- **Papier** : A4, carnet ; papier **à points ou à grille** pour aligner.
- **Feutres** :
  - noir **fin** pour le texte et les icônes ;
  - noir **épais** pour les titres et les conteneurs ;
  - **gris** pour le corps de texte (faible importance).
- **Post-its** : modales, menus déroulants, tooltips.
- **Ciseaux et ruban adhésif** : réarranger sans tout redessiner.



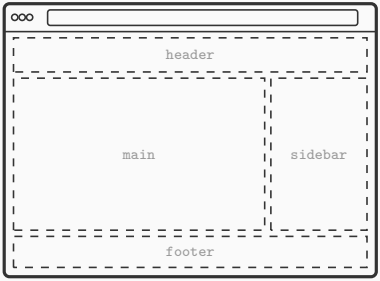
Le processus en 5 étapes

1. **Définir le cadre** : le *viewport* (écran de téléphone, fenêtre de navigateur) → contraintes réalistes.



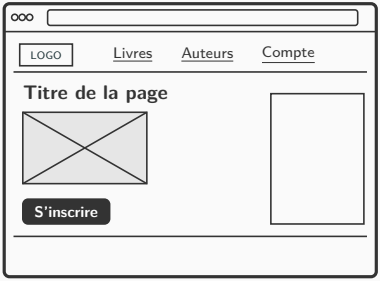
Le processus en 5 étapes

1. **Définir le cadre** : le *viewport* (écran de téléphone, fenêtre de navigateur) → contraintes réalistes.
2. **Placer les grands blocs** : `header`, `main`, `sidebar`, `footer`.



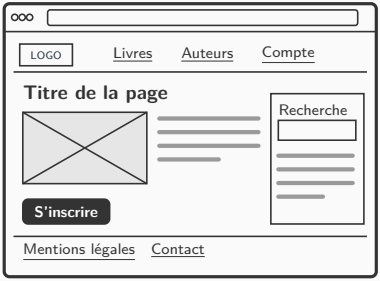
Le processus en 5 étapes

1. **Définir le cadre** : le *viewport* (écran de téléphone, fenêtre de navigateur) → contraintes réalistes.
2. **Placer les grands blocs** : `header`, `main`, `sidebar`, `footer`.
3. **Ajouter les éléments clés** : logo, navigation, titre, image héros, boutons d'appel à l'action (CTA).



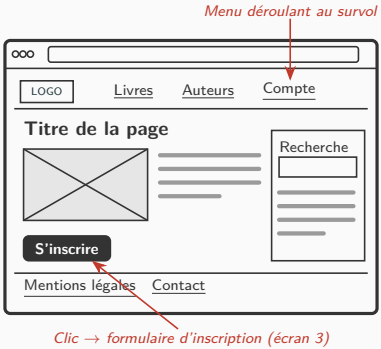
Le processus en 5 étapes

1. **Définir le cadre** : le *viewport* (écran de téléphone, fenêtre de navigateur) → contraintes réalistes.
2. **Placer les grands blocs** : `header`, `main`, `sidebar`, `footer`.
3. **Ajouter les éléments clés** : logo, navigation, titre, image héros, boutons d'appel à l'action (CTA).
4. **Remplir les détails** avec les conventions ; n'écrire que le texte **crucial** (labels, titres).



Le processus en 5 étapes

1. **Définir le cadre** : le *viewport* (écran de téléphone, fenêtre de navigateur) → contraintes réalistes.
2. **Placer les grands blocs** : `header`, `main`, `sidebar`, `footer`.
3. **Ajouter les éléments clés** : logo, navigation, titre, image héros, boutons d'appel à l'action (CTA).
4. **Remplir les détails** avec les conventions ; n'écrire que le texte **crucial** (labels, titres).
5. **Annoter !** Peut-être l'étape la plus importante : clics, états (`:hover`), logique invisible.



Conventions visuelles et notations

Adopter un **langage visuel commun** pour que les croquis soient compréhensibles par tous.

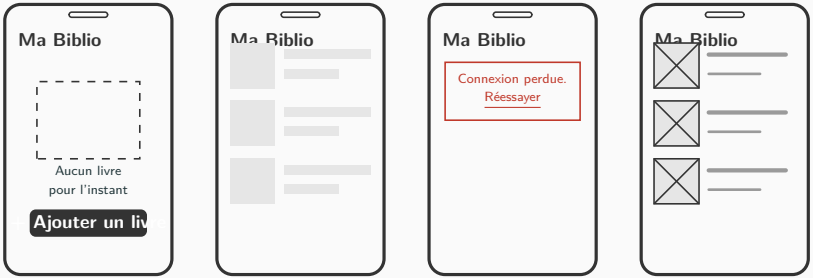
Élément	Dessin	Règle	Élément	Dessin	Règle
Image / média		rectangle barré	Bouton	S'inscrire	rectangle + label
Titre (H1, H2)	TITRE	majuscules / épais	Lien	En savoir plus	texte souligné
Paragraphe		lignes grises	Champ de saisie	Email	label au-dessus
Icône		forme simplifiée	Menu déroulant	Pays	triangle ▼

Astuce

Ne vous préoccupez pas d'être un grand artiste. La clarté prime sur l'esthétique. **Un carré est un carré.**

Ne pas oublier les états d'un écran

Un écran n'existe pas que dans son état « idéal ». Dessinez au moins les états qui changent l'interface :



L'état **vide** est souvent le **premier** que voit un nouvel utilisateur : c'est là qu'il faut le guider (*call to action*).

Simuler l'interaction : le test « guérilla »

Un prototype papier n'est pas statique : on peut faire des **tests utilisateurs** dès maintenant.

Les rôles

- **L'utilisateur** : un · e collègue à qui l'on donne une tâche (« Ajoute ce livre à ta bibliothèque »). Il « clique » du doigt.
- **L'ordinateur humain** : vous ! Vous remplacez la feuille par l'écran suivant, ou posez un post-it pour simuler une pop-up.
- **L'observateur** (optionnel) : note les hésitations, sans aider.

Règles d'or

- ne pas expliquer l'interface ;
- demander de **penser à voix haute** ;
- noter chaque hésitation ;
- 5 testeurs révèlent l'essentiel des problèmes majeurs.

Cette technique révèle très vite les problèmes de fluidité et de compréhension, **avant d'écrire une seule ligne de code**.

Le dessin : avantages et inconvénients

Avantages ✓

- **Rapidité imbattable** : idéal en brainstorming.
- **Coût zéro** : ni licence ni machine puissante.
- **Universel et inclusif** : tout le monde peut participer.
- **Zéro attachement** : on jette et on recommence.
- **Focus total** sur l'idée, pas sur l'outil.

Inconvénients ✗

- **Partage difficile** avec des équipes distantes.
- **Non durable** : le papier se perd, s'abîme.
- **Pas de réutilisation** : pas de « copier-coller » d'un bouton.
- Aspect « **non professionnel** » pour certains clients.
- **Pas de responsive** natif.

Méthodologie 2 : les outils numériques

Du papier aux pixels : la formalisation

Une fois les idées explorées sur papier, l'outil numérique permet de :

- **formaliser** la structure avec précision (alignements, espacements) ;
- **collaborer** efficacement avec l'équipe ;
- créer une **source unique de vérité** (*single source of truth*) ;
- **préparer** le prototypage interactif et le design haute fidélité.

Ces outils sont le **pont** entre l'idéation (Lo-Fi) et la maquette visuelle (Hi-Fi).

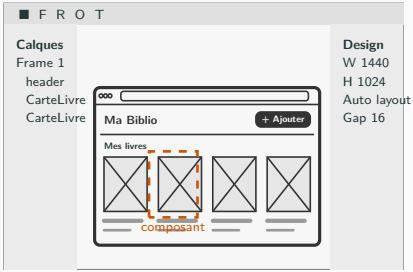
Focus : Figma, le standard de l'industrie

Qu'est-ce que Figma ?

- design d'interface et prototypage **dans le navigateur** (ou en appli) ;
- collaboration **en temps réel** via une simple URL ;
- offre gratuite et offre **Éducation** pour les étudiant · es.

Idéal pour le wireframing ?

- formes et textes en quelques touches ;
- alignement et distribution parfaits ;
- **composants** réutilisables.



Wireframer dans Figma : concepts clés

Les outils essentiels

- **Frame** (`F`) : vos écrans (*artboards*), avec des tailles prédéfinies (iPhone, Desktop…).
- **Formes** (`R` rectangle, `O` ellipse) : le rectangle est votre meilleur ami !
- **Texte** (`T`) : titres, labels, paragraphes.
- **Auto Layout** (`Maj+A`) : des conteneurs qui s'adaptent à leur contenu (l'équivalent de `flexbox`).

Le concept de composant (*Component*)

Créez un élément une fois (un bouton, une carte livre), transformez-le en composant (`Ctrl/Cmd+Alt+K`), puis réutilisez-le partout. Modifier le composant principal met à jour **toutes ses instances**. C'est la clé de la **cohérence** (cf. séquence 1) et de l'efficacité.

L'alternative libre : Penpot

La liberté de l'open source

Plateforme de design et de prototypage **open source**.

- **Gratuit** et libre ;
- **auto-hébergement** possible : contrôle total des données ;
- format basé sur le standard **SVG** ; mises en page proches de **CSS** (flex, grid) ;
- interface **familière** : la transition depuis Figma est aisée.

Autres outils

- **Balsamiq** : style volontairement « croquis », très rapide, payant ;
- **Excalidraw** : tableau blanc en ligne, idéal en réunion ;
- **draw.io** : diagrammes, gratuit.

Idéal pour les équipes qui privilégient un écosystème logiciel libre.

Outils numériques : avantages et inconvénients

Avantages ✓

- Précision et propreté.
- Collaboration en temps réel.
- Composants réutilisables.
- Partage facile (une URL).
- Prototypage cliquable.

Inconvénients ✗

- Le piège de la « **joliesse** » : on peaufine au lieu de structurer.
- Courbe d'apprentissage.
- Plus lent pour l'idéation.
- **Faux sentiment de finalité** : le client croit que c'est fini.

? Question à la salle

Pourquoi impose-t-on souvent les **niveaux de gris** pour un wireframe numérique ?

Méthodologie 3 : le wireframe en code

La maquette structurelle HTML/CSS

Définition

Construire la structure d'une page **directement en HTML** avec un minimum de CSS, pour établir le **squelette sémantique et structurel**. Objectif : valider la disposition dans l'environnement final, **le navigateur**.

C'est la méthode qui se rapproche le plus du produit final. Bien que plus lente, elle permet de :

- créer une fondation **sémantique et accessible** dès le début ;
- **tester le responsive** nativement (redimensionner la fenêtre suffit) ;
- **fluidifier la passe aux développeurs** : le code est un livrable utilisable ;
- **valider la faisabilité technique** immédiatement.

Le processus pratique

1. **Outils** : un éditeur de code et un navigateur.
2. **Squelette HTML** : dans `index.html`, les grandes balises sémantiques `<header>`, `<main>`, `<footer>`, `<nav>`, `<article>`.
3. **Révéler la structure** : dans `style.css`, des bordures pour voir les « boîtes ».
4. **Contenu factice** (*placeholder*) : titres, paragraphes, images (`placehold.co`).

Le « debug CSS »

```
* { border: 1px solid red; }
```

Une ligne pour voir comment **chaque** élément occupe l'espace.

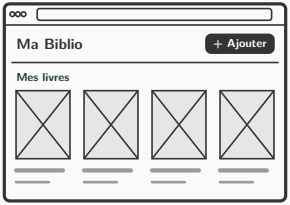
Images factices

```
<img src="https://placehold.co/100x150"
     alt="Couverture de Dune">
```

Exemple : l'écran « Bibliothèque » du TP

```html
<body>
  <header>
    <h1>Ma Biblio</h1>
    <button>+ Ajouter un livre</button>
  </header>
  <main>
    <h2>Mes livres</h2>
    <div class="book-list">
      <article class="book-item">
        <img src="https://placehold.co/100
     x150"
             alt="Couverture du livre 1">
        <h3>Titre du livre 1</h3>
        <p>Nom de l'auteur</p>
      </article>
      <!-- ... autres livres ... -->
    </div>
  </main>
</body>
```

```css
/* Style minimaliste */
body { font-family: sans-
      serif; }
header, main {
  border: 2px solid #333;
  padding: 1rem; margin: 1rem
      ;
}
.book-list {
  border: 1px dotted #666;
  padding: 1rem;
  display: flex;
  gap: 1rem;
}
.book-item {
  border: 1px solid #ccc;
  padding: .5rem;
  text-align: center;
}
```



Question : que se passe-t-il avec 30 livres et `display:flex` ? Comment corriger ? (`flex-wrap` ou `grid`)

Wireframe en code : avantages et inconvénients

Avantages ✓

- Le plus **réaliste**.
- **Responsive** natif.
- Base réelle pour le développement.
- Encourage la **sémantique** (et donc l'accessibilité).
- Aucun logiciel spécifique requis.

Inconvénients ✗

- **Lenteur** pour explorer plusieurs idées.
- Demande des compétences techniques.
- **Rigidité** : on hésite à jeter du code.
- Intimidant pour les non-techniciens.

Synthèse et mise en pratique

Quelle méthode, quand ?

	Papier	Outil numérique	Code HTML/CSS
Vitesse d'exploration	★★★	★★	★
Précision / propreté	★	★★★	★★
Partage, collaboration	★	★★★	★★
Réalisme, responsive	★	★★	★★★
Accessible à tous	★★★	★★	★
Moment idéal	idéation, atelier	documentation, validation	passe aux devs

À retenir

Ce ne sont pas des méthodes concurrentes mais des **étapes** : on part du papier, on formalise en numérique, on valide en code.

Les erreurs les plus fréquentes

- Trop de détails trop tôt (couleurs, polices, logo soigné).
- Du *lorem ipsum* **partout**, même sur les boutons : « Lorem » n'est pas un label !
- Pas d'**annotations** : personne ne comprend ce qui se passe au clic.
- Seulement la version **desktop**, ou seulement l'état « rempli ».
- Des écrans isolés, sans **parcours** entre eux.
- Oublier les lois : 15 entrées de menu (**Hick**), CTA minuscule (**Fitts**), panier en bas à gauche (**Jakob**).

? Question à la salle

Un client vous dit, en voyant votre wireframe gris : « C'est triste, on peut mettre du bleu ? ». Que lui répondez-vous ?

Mise en pratique

TD (1h15) – papier

1. Déconstruire une page produit e-commerce.
2. Hiérarchiser le contenu d'une page « Nous contacter ».
3. La décliner en version mobile.

TP (1h15) – « Ma Biblio »

Les wireframes d'une application de gestion de livres, en passant par les **3 méthodologies** :

1. 3 écrans sur papier + test guérilla ;
2. 1 écran en Mid-Fi dans Figma/Penpot ;
3. l'écran d'accueil en HTML/CSS.

À apporter : crayon, feutre noir, compte Figma (ou Penpot) créé **avant** le TP.

Questions ?