

Création de Wireframes : Méthodologies et Outils

PV with the help of Gemini 2.5 Pro

IUT Fontainebleau

17 septembre 2025

Plan du cours

- 1 Introduction et Concepts Fondamentaux
- 2 Méthodologies et Bonnes Pratiques
- 3 Exercices (TD & TP)

Qu'est-ce qu'un Wireframe ?

Définition

Un **wireframe** (ou maquette fil de fer) est une représentation visuelle schématique de la structure d'une page web ou d'une 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 principal : l'architecture de l'information.

Pourquoi faire des Wireframes ?

Communication

- Aligner la vision entre les designers, développeurs et clients.
- Fournir une base de discussion concrète.

Efficacité

- Économiser du temps et de l'argent.
- "Échouer vite pour réussir plus tôt."
- Sert de plan pour les UI designers et les développeurs.

Validation

- Tester les parcours utilisateurs très tôt.
- Itérer rapidement et à faible coût.

Niveaux de Fidélité (Fidelity)

Low-Fidelity (Lo-Fi)

Dessins rapides sur papier ou tableau blanc. Idéal pour le brainstorming et l'exploration d'idées.

Mid-Fidelity (Mid-Fi)

Créés avec des outils numériques (ex : Figma, Balsamiq). En niveaux de gris, avec des blocs et du texte standard. Le plus courant pour la documentation.

High-Fidelity (Hi-Fi)

Très détaillé, proche du produit final (parfois appelé "mockup"). Inclut des éléments visuels (couleurs, typographies). Souvent utilisé pour les prototypes interactifs avant le développement.

Méthodologie 1 : Le Dessin (Paper Prototyping)

Le Cerveau, le Papier, le Crayon

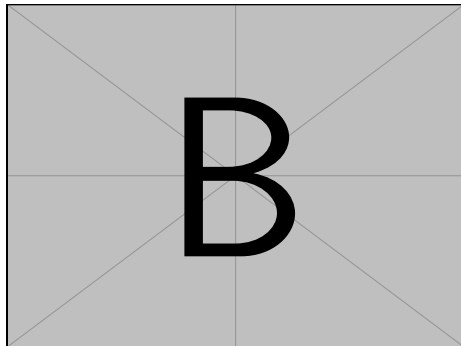
La méthode la plus fondamentale, la plus rapide et souvent la plus efficace pour passer d'une idée abstraite à un concept tangible.

"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 Kit de base :

- **Papier** : Feuilles A4, carnets. Le papier à grille ou à points est excellent pour l'alignement.
- **Outils d'écriture** :
 - Un feutre noir fin pour le texte et les icônes.
 - Un feutre noir épais pour les titres et les conteneurs.
 - Un feutre gris pour le corps du texte (simule une faible importance).
- **Post-its** : Parfaits pour simuler des modales, des menus déroulants ou des tooltips.
- **Ciseaux et ruban adhésif** : Pour découper et réarranger des éléments sans tout redessiner.



Votre poste de travail pour le paper prototyping

Le Processus en 5 Étapes

- 1 **Définir le cadre** : Dessinez un rectangle représentant le "viewport" (l'écran d'un téléphone, la fenêtre d'un navigateur). Cela vous donne des contraintes réalistes.

Le Processus en 5 Étapes

- 1 **Définir le cadre** : Dessinez un rectangle représentant le "viewport" (l'écran d'un téléphone, la fenêtre d'un navigateur). Cela vous donne des contraintes réalistes.
- 2 **Placer les grands blocs** : Identifiez et dessinez les zones structurelles principales : en-tête ('header'), zone de contenu principal ('main'), barre latérale ('sidebar'), pied de page ('footer').

Le Processus en 5 Étapes

- ❶ **Définir le cadre** : Dessinez un rectangle représentant le "viewport" (l'écran d'un téléphone, la fenêtre d'un navigateur). Cela vous donne des contraintes réalistes.
- ❷ **Placer les grands blocs** : Identifiez et dessinez les zones structurales principales : en-tête ('header'), zone de contenu principal ('main'), barre latérale ('sidebar'), pied de page ('footer').
- ❸ **Ajouter les éléments clés** : À l'intérieur des grands blocs, placez les composants importants : logo, navigation principale, titre de la page, image héros, boutons d'appel à l'action (CTA), etc.

Le Processus en 5 Étapes

- ❶ **Définir le cadre** : Dessinez un rectangle représentant le "viewport" (l'écran d'un téléphone, la fenêtre d'un navigateur). Cela vous donne des contraintes réalistes.
- ❷ **Placer les grands blocs** : Identifiez et dessinez les zones structurelles principales : en-tête ('header'), zone de contenu principal ('main'), barre latérale ('sidebar'), pied de page ('footer').
- ❸ **Ajouter les éléments clés** : À l'intérieur des grands blocs, placez les composants importants : logo, navigation principale, titre de la page, image héros, boutons d'appel à l'action (CTA), etc.
- ❹ **Remplir les détails** : Utilisez les conventions visuelles (voir diapo suivante) pour le texte, les images, les champs de formulaire. N'écrivez que le texte crucial (labels de boutons, titres).

Le Processus en 5 Étapes

- ➊ **Définir le cadre** : Dessinez un rectangle représentant le "viewport" (l'écran d'un téléphone, la fenêtre d'un navigateur). Cela vous donne des contraintes réalistes.
- ➋ **Placer les grands blocs** : Identifiez et dessinez les zones structurelles principales : en-tête ('header'), zone de contenu principal ('main'), barre latérale ('sidebar'), pied de page ('footer').
- ➌ **Ajouter les éléments clés** : À l'intérieur des grands blocs, placez les composants importants : logo, navigation principale, titre de la page, image héros, boutons d'appel à l'action (CTA), etc.
- ➍ **Remplir les détails** : Utilisez les conventions visuelles (voir diapo suivante) pour le texte, les images, les champs de formulaire. N'écrivez que le texte crucial (labels de boutons, titres).
- ➎ **Annoter !** C'est peut-être l'étape la plus importante. Utilisez des flèches et de petites notes pour expliquer ce qui se passe lorsqu'on clique sur un élément, les états (ex : :hover), ou toute logique non visible.

Conventions Visuelles et Notations

Il est utile d'adopter un langage visuel cohérent pour que vos croquis soient compréhensibles par tous.

Élément	Représentation Commune
Image / Média	Rectangle avec une grande croix à l'intérieur.
Titre (H1, H2...)	Texte court et lisible, souvent en majuscules ou plus épais.
Paragraphe de texte	Lignes ondulées ou ensemble de lignes droites.
Bouton	Rectangle avec un label clair (ex : "S'inscrire").
Lien hypertexte	Texte souligné.
Champ de saisie	Rectangle vide, avec un label au-dessus.
Menu déroulant	Rectangle avec un petit triangle pointant vers le bas.
Icône	Représentation simplifiée (une loupe pour la recherche, etc.)

Astuce

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

Simuler l'Interaction : Le Test "Guérilla"

Un prototype papier n'est pas statique ! Il peut être utilisé pour effectuer des tests utilisateurs très tôt.

Les Rôles :

- **L'Utilisateur** : Une personne (un collègue, un ami) à qui l'on donne une tâche à accomplir (ex : "Ajoute ce livre à ton panier"). L'utilisateur "clique" en pointant du doigt les éléments.
- **L'Ordinateur Humain** : C'est vous ! Lorsque l'utilisateur clique sur un élément interactif, vous réagissez en remplaçant la feuille de papier par la nouvelle page correspondante, ou en ajoutant un Post-it pour simuler une pop-up.

Cette technique révèle très vite les problèmes de fluidité et de compréhension dans vos parcours utilisateurs, avant même d'avoir écrit une seule ligne de code !

Avantages et Inconvénients du Dessin

Avantages

- **Rapidité imbattable** : Idéal pour le brainstorming.
- **Coût zéro** : Pas besoin de licence ou d'ordinateur puissant.
- **Universel et inclusif** : Tout le monde peut prendre un crayon et participer.
- **Zéro attachement** : On n'hésite pas à jeter une feuille et à recommencer.
- **Focus total sur l'idée** : Aucune distraction liée à l'outil.

Inconvénients

- **Partage difficile** : Complexe à présenter à des équipes distantes.
- **Non durable** : Le papier se perd, s'abîme.
- **Pas de réutilisabilité** : On ne peut pas "copier-coller" un bouton.
- **Aspect "non professionnel"** : Peut nécessiter une explication pour certains clients.
- **Pas de responsive natif.**

Du Papier aux Pixels : la Formalisation

Une fois les idées initiales explorées sur papier, les outils numériques permettent de :

- **Formaliser** la structure avec précision.
- **Collaborer** efficacement avec les autres membres de l'équipe.
- **Créer une source unique de vérité** ('single source of truth').
- **Préparer** le terrain pour le prototypage interactif et le design haute-fidélité.

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

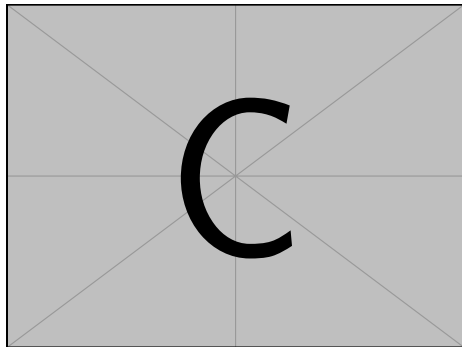
Focus : Figma (Le Standard de l'Industrie)

Qu'est-ce que Figma ?

- Un outil de design d'interface et de prototypage **basé sur le navigateur**.
- Accès et collaboration en temps réel via une simple URL.
- Son **plan gratuit est extrêmement généreux**.

Idéal pour le wireframing ?

- Facilité de création de formes et de textes.
- Fonctions d'alignement et de distribution parfaites.
- Possibilité de créer des composants réutilisables.



Interface de Figma avec un wireframe mid-fidelity

Wireframing dans Figma : Concepts Clés

Les Outils Essentiels

- **Frames (raccourci F)** : Ce sont vos écrans ou "artboards".
- **Outils de forme (raccourcis R, O)** : Le rectangle est votre meilleur ami !
- **Outil Texte (raccourci T)** : Pour les titres, labels et paragraphes.
- **Auto Layout** : Pour créer des conteneurs qui s'adaptent à leur contenu.

Le concept de Composant (*Component*)

Créez un élément une fois (ex : un bouton), transformez-le en composant, puis réutilisez-le partout. Si vous modifiez le composant principal, toutes ses copies (instances) se mettent à jour. C'est la clé de la cohérence et de l'efficacité.

Penpot

La Liberté de l'Open Source

Penpot est la première plateforme de design et de prototypage Open Source.

Avantages spécifiques :

- **Gratuit et Open Source.**
- **Auto-hébergement possible** pour un contrôle total.
- **Format basé sur le standard SVG**, un standard du web.
- **Interface familière** : La transition depuis Figma est aisée.

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.
- Prototypage.

Inconvénients 👎

- Le piège de la "joliessse".
- Courbe d'apprentissage.
- Plus lent pour l'idéation.
- Faux sentiment de finalité.

Définition : La Maquette Structurale HTML/CSS

Il s'agit de construire la structure d'une page web directement avec du code HTML et un minimum de CSS pour établir le **squelette sémantique et structurel**. L'objectif est de valider la disposition dans l'environnement final : **le navigateur web**.

C'est la méthode de wireframing qui se rapproche le plus du produit final.

Quand et Pourquoi Choisir Cette Méthode ?

Bien que plus lente, cette approche est précieuse pour :

- **Créer une fondation sémantique et accessible** dès le début.

Quand et Pourquoi Choisir Cette Méthode ?

Bien que plus lente, cette approche est précieuse pour :

- **Créer une fondation sémantique et accessible** dès le début.
- **Tester le responsive design nativement.**

Quand et Pourquoi Choisir Cette Méthode ?

Bien que plus lente, cette approche est précieuse pour :

- **Créer une fondation sémantique et accessible** dès le début.
- **Tester le responsive design nativement.**
- **Fluidifier la passe aux développeurs** : Le code est un livrable utilisable.

Quand et Pourquoi Choisir Cette Méthode ?

Bien que plus lente, cette approche est précieuse pour :

- **Créer une fondation sémantique et accessible** dès le début.
- **Tester le responsive design nativement.**
- **Fluidifier la passe aux développeurs** : Le code est un livrable utilisable.
- **Valider la faisabilité technique** immédiatement.

- ❶ **Outils** : Un éditeur de code et un navigateur web.
- ❷ **Créer le squelette HTML** : Dans 'index.html', placez les grandes balises sémantiques : '<header>', '<main>', '<footer>'.
- ❸ **Révéler la structure avec CSS** : Dans 'style.css', utilisez des bordures pour visualiser les "boîtes".

Le "Debug CSS"

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

Cette ligne de CSS est un outil formidable pour voir comment chaque élément occupe l'espace.

- ❹ **Remplir avec du contenu placeholder** : Ajoutez les titres, paragraphes et images.

Exemple : Écran "Bibliothèque" du TP

Voici à quoi pourrait ressembler le code pour la page d'accueil de notre TP "Ma Biblio".

```
<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">
        
        <h3>Titre du Livre 1</h3>
        <p>Nom de l'Auteur</p>
      </article>

      <article class="book-item">
        
        <h3>Autre Titre</h3>
        <p>Autre Auteur</p>
      </article>
    </div>
  </main>
</body>
```

```
/* 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: 0.5rem;
  text-align: center;
}
```

Wireframe en Code : Avantages et Inconvénients

Avantages

- Le plus réaliste.
- Responsive natif.
- Base pour le développement.
- Encourage la sémantique.
- Aucun logiciel requis.

Inconvénients

- Lenteur.
- Compétences techniques.
- Rigidité.
- Intimidant pour non-tech.

Exercices Dirigés (TD)

Nous allons réaliser deux petits exercices pour nous familiariser avec les concepts :

- 1 Déconstruction d'un site existant sur papier.
- 2 Organisation du contenu d'une page.

Travaux Pratiques (TP)

Le TP final consistera à créer les wireframes pour une petite application de gestion de livres, en passant par les 3 méthodologies vues en cours.

Questions ?