

## 1. Conception responsive

- Le site web doit s'adapter parfaitement à différents formats d'écran (ordinateur, tablette, mobile).
- Utilisez des media queries CSS, des grilles flexibles et/ou des frameworks comme Bootstrap ou Tailwind CSS.

## 2. Visuellement attrayant

- Utilisez efficacement les couleurs, les polices, les images et la mise en page.
- Assurez-vous que le contenu est lisible et engageant.
- Évitez l'encombrement et maintenez un aspect professionnel.

## 3. Barre de menu avec en-tête et pied de page

- Incluez un **en-tête** avec le logo ou le titre de votre site ainsi qu'un menu de navigation.
- Ajoutez un **pied de page** contenant des informations de contact, un avis de droits d'auteur et des liens utiles.

## 4. Intégration audio et vidéo

- Intégrez au moins un **fichier audio** (ex : musique, narration).
- Intégrez au moins **une vidéo** (ex : tutoriel, présentation).
- Assurez-vous que les fichiers multimédias fonctionnent sur tous les navigateurs principaux.

## 5. Page d'inscription et de connexion

- Créez des pages **d'authentification utilisateur**.
- Ajoutez une validation des champs pour le nom d'utilisateur/email et le mot de passe.
- Stockez les identifiants de manière sécurisée (hash des mots de passe recommandé).

## 6. Page dynamique pour les utilisateurs

- Créez au moins **une page affichant du contenu dynamique** en fonction de l'utilisateur ou de ses actions.
- Exemples : message personnalisé, tableau de bord, données spécifiques à l'utilisateur.

## 7. Cookies et sessions

- Implémentez des cookies ou une gestion de session pour :
  - Mémoriser les utilisateurs connectés.
  - Enregistrer les préférences de l'utilisateur.

- Montrez une gestion correcte de la création, de l'utilisation et de l'expiration des cookies/sessions.

## 8. Gestion et nommage appropriés des fichiers

- Organisez vos fichiers de manière claire (ex : `index.html`, `css/styles.css`, `js/scripts.js`).
- Utilisez des noms de fichiers et dossiers significatifs.
- Maintenez une structure cohérente.

## 9. Page README pour Git

- Ajoutez un fichier `README.md` contenant :
  - Une description du projet
  - Les fonctionnalités
  - Les instructions pour exécuter le site en local
  - Les outils et technologies utilisés
  - Les défis rencontrés ou fonctionnalités supplémentaires

## 10. Dépôt GitHub

- Téléversez votre projet dans un **dépôt GitHub**.
- Utilisez des messages de commit appropriés et clairs.
- Le dépôt doit être public ou partagé avec l'enseignant.