

Tutoriel – Créer une page Web de présentation d'un kit graphique

Titre RNCP : DWWM – Développeur Web et Web Mobile

Technologies utilisées : HTML5, CSS3, JavaScript ES2023+

Objectif

Créer une page web simple, responsive et accessible permettant de présenter le kit graphique d'un projet.

Étape 1 : Créer l'arborescence du projet

Dans votre projet, créer l'arborescence suivante :

```
src/assets/  
|—— kit-vore-nom.html  
|—— kit-vore-nom.css
```

✅ Vérifiez que tous les répertoires et fichiers demandés dans le ticket sont créés.

Étape 2 : Créer la structure HTML

Dans `kit-vore-nom.html`, ajoutez les grandes zones de votre page :

```
<header></header>  
<nav></nav>  
<main>  
  <section id="couleurs"></section>  
  <section id="typographies"></section>  
  <section id="composants"></section>  
  <section id="icones"></section>  
</main>  
<footer>Nom du projet - Nom de l'équipe de développement</footer>
```

✅ Utilisez uniquement des balises sémantiques.

Étape 3 : Ajouter l'identité du projet

Dans le header :

- Ajouter le logo.
- Ajouter le nom du projet.
- Ajouter une courte description.

Exemple :

```
<header>
  
  <h1>Kit Graphique NomDuProjet</h1>
  <p>Référence visuelle du projet.</p>
</header>
```

✓ Le logo doit posséder un texte alternatif.

Étape 4 : Présenter la palette de couleurs

Créer une section affichant plusieurs couleurs.

Exemple :

```
<div class="color-card">
  <div class="color-preview"></div>
  <p>#2563EB</p>
</div>
```

```
.color-preview {
  width: 120px;
  height: 120px;
  background: #2563eb;
}
```

✓ Afficher le nom, le code HEX et éventuellement le code RGB.

Étape 5 : Présenter les typographies

Sélectionner une ou deux polices.

Exemple :

```
<section id="typographies">
  <h2>Typographies</h2>
  <h3>Titre avec la police principale</h3>
  <p>Exemple de texte courant.</p>
</section>
```

✅ Indiquer l'usage de chaque police.

Étape 6 : Créer les composants graphiques

Présenter :

- Bouton principal
- Bouton secondaire
- Champ de formulaire
- Carte produit ou service

Exemple :

```
<button class="btn-primary">Valider</button>
<button class="btn-secondary">Annuler</button>
```

```
.btn-primary {
  background: #2563eb;
  color: white;
}
```

✅ Tous les composants doivent respecter la même charte graphique.

Étape 7 : Ajouter les icônes et visuels

Créer une section contenant :

- Les icônes principales.
- Les illustrations du projet.
- Les règles d'utilisation.

✓ Organiser les visuels dans une grille.

Étape 8 : Mettre en page avec CSS Grid ou Flexbox

Exemple :

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));  
  gap: 1rem;  
}
```

✓ Vérifier que les blocs s'adaptent automatiquement à la largeur disponible.

Étape 9 : Rendre la page responsive

Ajouter une media query.

```
@media (min-width: 768px) {  
  nav {  
    /* code CSS */  
  }  
}
```

✓ Tester sur :

- Mobile
- Tablette
- Ordinateur

Étape 10 : Vérifier l'accessibilité

Contrôler :

- Les contrastes.
- Les textes alternatifs.
- La hiérarchie des titres.
- L'utilisation du clavier.

✓ La page doit rester utilisable sans souris.

Étape 11 : Finaliser le projet

Avant la livraison :

- Valider le HTML.
 - Vérifier le CSS.
 - Corriger les erreurs.
 - Compléter le README principal du projet.
-

Critères de réussite

Niveau attendu

- Structure HTML claire.
 - Responsive design fonctionnel.
 - Palette cohérente.
 - Composants homogènes.
 - Respect des bonnes pratiques d'accessibilité.
-

Bonus

- Utiliser des variables CSS.
- Ajouter un mode sombre.
- Ajouter quelques animations CSS.
- Créer un sélecteur de thème avec JavaScript (plus tard !!!).

Compétences DWM mobilisées

- Maquetter une application.
- Réaliser une interface utilisateur web statique et adaptable.
- Développer la partie front-end d'une application web.
- Appliquer les bonnes pratiques de qualité et d'accessibilité.

Compétences CDA mobilisées

- Maquetter une application.
- Développer une interface utilisateur.