

## TP6 : Site responsive avec CSS Grid

Cours HTML/CSS 2026 – Durée : 3h30 – Enseignant : Quentin Fait

---

### Objectifs du TP

- Créer un layout de page avec CSS Grid
  - Utiliser les media queries pour adapter le design
  - Combiner Grid et Flexbox
  - Produire un site responsive complet
- 

### Consignes générales

**Configuration :** Créez un dossier `tp6-grid-responsive` sur votre ordinateur et ouvrez-le dans votre éditeur de code.

**Important :** Testez en redimensionnant votre fenêtre à chaque étape pour voir l'adaptation.

---

### Exercice : Site de recettes de cuisine

**Fichiers :** `recettes.html`, `recettes.css`, et 6 images de plats

Créez un site de recettes responsive qui présente différents plats.

### Préparation des images

- Trouvez 6 images de plats cuisinés (ou utilisez des images de placeholder)
  - Nommez-les `plat1.jpg` à `plat6.jpg`
  - Placez-les dans le dossier `tp6-grid-responsive`
- 

### Partie HTML (`recettes.html`)

Structure de base fournie

Récupérez le fichier HTML de départ : `recettes.html`

Le HTML est complet.

---

## Partie CSS (recettes.css)

C'est vous qui allez écrire tout le CSS en suivant les étapes.

### Étape 1 : Reset et base

Copiez-collez ces premiers éléments dans le fichier `recettes.css` :

```
* {  
    margin: 0;  
    padding: 0;  
    box-sizing: border-box;  
}  
  
body {  
    font-family: Arial, sans-serif;  
    color: #333;  
}  
  
img {  
    max-width: 100%;  
    height: auto;  
}
```

---

### Étape 2 : Header avec Flexbox

**Objectif :** Header avec titre à gauche, navigation à droite

Ciblez `site-header` et appliquez : - Fond : `#c0392b` - Texte blanc - Padding : 20px - Ombre : `box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1)`;

Ciblez `header-content` et créez un layout Flexbox : - `display: flex`; - Espacer titre et nav : `justify-content: space-between`; - Aligner verticalement : `align-items: center`; - Largeur max : 1200px - Centrage : `margin: 0 auto`;

Ciblez le `<h1>` dans le header : - Taille : 28px

Ciblez `main-nav` et créez une navigation horizontale : - `display: flex`; - Espacement : `gap: 25px`;

Ciblez les liens dans `main-nav` : - Texte blanc - Pas de soulignement - Padding : 8px 12px - Transition : 0.3s sur `background-color`

Au survol, ajoutez un fond : `background-color: rgba(0, 0, 0, 0.2)`;

---

### Étape 3 : Layout principal avec Grid (mobile d'abord)

#### Sur mobile : tout empilé verticalement

Ciblez `page-layout` et appliquez : - `display: grid;` - Une seule colonne : `grid-template-columns: 1fr;` - Hauteur minimale : `min-height: calc(100vh - 150px);`

---

### Étape 4 : Sidebar

Ciblez `sidebar` et appliquez : - Fond : `#f8f9fa` - Padding : 20px - Bordure droite : 1px solid `#ddd`

Ciblez les `<h3>` dans `sidebar` : - Taille : 18px - Couleur : `#2c3e50` - Marge basse : 15px

Ciblez `search-box` : - Marge basse : 30px

Ciblez les inputs dans `search-box` : - Largeur : 100% - Padding : 10px - Bordure : 1px solid `#ddd` - Coins arrondis : 4px - Marge basse : 10px

Ciblez les boutons dans `search-box` : - Largeur : 100% - Padding : 10px - Fond : `#c0392b` - Texte blanc - Bordure : none - Coins arrondis : 4px - Cursor : pointer

Au survol, changez le fond en `#a93226`

Ciblez les `<ul>` dans `categories` : - `list-style: none;`

Ciblez les `<li>` dans `categories` : - Padding : 8px 0 - Bordure basse : 1px solid `#e0e0e0` - Cursor : pointer

Au survol des `<li>`, changez la couleur en `#c0392b`

---

### Étape 5 : Contenu principal

Ciblez `main-content` et appliquez : - Padding : 20px

---

### Étape 6 : Bannière hero

Ciblez `hero-banner` et créez une bannière centrée : - Fond : dégradé `linear-gradient(135deg, #667eea, #764ba2);` - Texte blanc - Padding : 60px 20px - Alignement centré - Coins arrondis : 8px - Marge basse : 30px

Ciblez le `<h2>` dans `hero-banner` : - Taille : 32px - Marge basse : 10px

Ciblez le `<p>` dans `hero-banner` : - Taille : 18px - Opacité : 0.9

---

## Étape 7 : Grille de recettes (mobile)

### Mobile : 1 colonne

Ciblez `recipes-grid` et créez une grille : - `display: grid;` - Une colonne : `grid-template-columns: 1fr;` - Espacement : `gap: 20px;`

---

## Étape 8 : Cartes de recettes

Ciblez `recipe-card` et créez des cartes : - Fond blanc - Bordure : `1px solid #ddd` - Coins arrondis : `8px` - Overflow : `hidden` - Ombre au survol : `box-shadow: 0 4px 12px rgba(0, 0, 0, 0.15);` sur `:hover` - Transition : `0.3s` sur `box-shadow`

Ciblez les images dans `recipe-card` : - Largeur : `100%` - Hauteur : `200px` - `object-fit: cover;`

Ciblez `recipe-info` et créez un conteneur Flexbox : - Padding : `15px` - `display: flex;` - `flex-direction: column;` - `gap: 8px;`

Ciblez les `<h3>` dans `recipe-info` : - Taille : `20px` - Couleur : `#2c3e50`

Ciblez `time` et `difficulty` : - Taille : `14px` - Couleur : `#7f8c8d`

---

## Étape 9 : Footer

Ciblez `site-footer` et créez un footer centré : - Fond : `#2c3e50` - Texte blanc - Padding : `20px` - Alignement centré

---

## Étape 10 : Media query Tablette (768px)

### À partir de 768px : 2 colonnes pour les recettes

```
@media (min-width: 768px) {  
  .recipes-grid {  
    grid-template-columns: repeat(2, 1fr);  
  }  
}
```

---

## Étape 11 : Media query Desktop (1024px)

### À partir de 1024px : sidebar à gauche + 3 colonnes de recettes

```
@media (min-width: 1024px) {  
  /* Layout avec sidebar */  
  .page-layout {
```

```

        grid-template-columns: 280px 1fr;
        max-width: 1400px;
        margin: 0 auto;
    }

    /* Contenu principal */
    .main-content {
        padding: 30px;
    }

    /* Hero banner plus grand */
    .hero-banner {
        padding: 80px 40px;
    }

    .hero-banner h2 {
        font-size: 42px;
    }

    .hero-banner p {
        font-size: 22px;
    }

    /* Grille 3 colonnes */
    .recipes-grid {
        grid-template-columns: repeat(3, 1fr);
        gap: 25px;
    }

    /* Navigation horizontale plus espacée */
    .main-nav {
        gap: 30px;
    }
}

```

---

## Rendu attendu

Votre site doit s'adapter ainsi :

**Mobile (< 768px) :** - Header avec nav empilée sous le titre - Sidebar en haut (pleine largeur) - Grille de recettes : 1 colonne - Tout empilé verticalement

**Tablette (768px - 1024px) :** - Header avec nav à droite - Sidebar toujours en haut - Grille de recettes : 2 colonnes

**Desktop (> 1024px) :** - Header avec nav à droite - Sidebar fixe à gauche

(280px) - Contenu principal à droite - Grille de recettes : 3 colonnes - Bannière hero plus grande

---

## Points de vérification

Votre site utilise : - **Grid** pour le layout principal (sidebar + contenu) - **Grid** pour la grille de recettes adaptative - **Flexbox** pour le header et la navigation - **Flexbox** pour les infos dans chaque carte - **3 media queries** : une pour tablette, une pour desktop - **Balise viewport** dans le HTML

Testez en redimensionnant la fenêtre du navigateur

---

## Livrables attendus

Un dossier **NOM-tp6** contenant : - **recettes.html** (fourni) - **recettes.css** (créé par vous) - 6 images de plats

Pensez à valider votre code CSS avec le validateur W3C : <https://jigsaw.w3.org/css-validator/>

---

## Critères d'évaluation

- **CSS Grid** : layout principal et grille de recettes
  - **Flexbox** : header, navigation, cartes
  - **Media queries** : adaptation mobile/tablette/desktop
  - **Responsive** : le site s'adapte correctement à toutes les tailles
  - **Code propre** : bien indenté, commenté si nécessaire
  - **Validation W3C** : CSS valide
- 

## Bonus (optionnel)

Si vous finissez en avance, ajoutez ces améliorations :

1. **Effet au survol sur les cartes** : translation vers le haut

```
.recipe-card:hover {  
  transform: translateY(-5px);  
}
```

2. **Navigation responsive** : sur mobile, cachez la nav et affichez un bouton menu
3. **Grille avec auto-fit** : remplacez les media queries de la grille par :

```
.recipes-grid {  
  grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));  
}
```

---