

## TP 1 – 2 : Dégradation responsive

*Chers étudiants :*

1. *Si vous avez une tablette personnelle qu'il vous est possible d'amener au 2<sup>nd</sup> TP, elle sera la bienvenue !*
2. *Entre les 2 TP, du travail personnel vous sera demandé pour une bonne finalisation du projet. (voir ci-dessous, p1, « progression sur les 2 TP »)*
3. *A des fins pédagogiques, dans ce TP, nous aborderons la dégradation responsive. En TP devoir, sauf changement, il vous sera demandée une approche mobile first.*

### Objectif final

- ✓ Créer une page responsive webdesign en appliquant le principe de la dégradation responsive.

### Objectifs opérationnels

- ✓ Réaliser une mise en page avec flexbox
- ✓ Ecrire des media queries
- ✓ Gérer des images dans un contexte « responsive »
- ✓ Gérer du texte et une grille fluides

### Durée

- ✓ 2 TP soit 6h00 + travail personnel

### Outils

- ✓ Un éditeur web : Brackets / sublimeText / Atom / notepad++ / ...
- ✓ Navigateurs *chrome* ou *Firefox* de préférence !
- ✓ *Inspecteur d'éléments* du navigateur (clic droit sur la page depuis le navigateur)
- ✓ La *vue adaptative* du navigateur :
  - depuis *Firefox* : clic droit + inspecter l'élément + clic sur icône « vue adaptative »
  - depuis *Chrome* : F12 + clic sur icône « vue adaptative »
  - depuis *Safari* : Menu «Préférences» -> onglet «avancé» -> cocher option «développement» tout en bas de la fenêtre si ce n'est déjà fait. L'option « développement » apparaît alors dans le menu, la dérouler et choisir « vue adaptative »

### Ressources fournies

- ✓ Les ressources nécessaires à l'intégration (maquette, images, fonts, ...) sont en ligne sur CELENE : <http://celene.univ-tours.fr/> -> IUT Blois -> S3 -> MMI-S3-Integration multimédia -> TP 1-2

### Progression sur les 2 TPs

- ✓ TP1 (3h00) : Etapes 1 à 2 incluses
- ✓ Entre les 2 TP : Etape 3
- ✓ TP2 (3h00) : Etape 4 à 5
- ✓ Remarque : en plus de l'étape3, vous devrez avancer entre les 2 tp si votre progression a été moins rapide que celle proposée.

### Evaluation

- ✓ Le TP sera ramassé, susceptible d'être noté
- ✓ Toute absence non justifiée = 0
- ✓ Une note d'investissement est attribuée à chaque TP

## TP 1 – 2 : Dégradation responsive

---

### ETAPE 1 : Analyse des mises en page fournies et écriture du code html (20-30min)

On souhaite rendre une page responsive en procédant à de la dégradation responsive. Deux mises en page ont été créées :

- ✓ celle de la page web initiale (desktop). Voir *maquette\_medium.jpg ou annexe*. Mise en page pour les résolutions supérieures à 900px.
  - ✓ une pour les résolutions inférieures à 900px. Voir *zoning\_tiny.xls*.  
C'est une réorganisation des éléments de la page précédente pour une mise en page pour petites résolutions.
1. Analysez et notez les différences de positionnement entre ces 2 mises en page. Dessinez le zoning de la maquette « desktop » (tenez compte des 2 mises en page pour faciliter votre travail d'intégration).
  2. Ecrivez le code html de la page en respectant la sémantique (balises : header / section / article / nav / footer / div / span / ...). Vous avez dû remarquer à l'étape précédente que les mises en pages ne présentent pas de grandes différences : les éléments sont quasi inchangés d'une mise en page à l'autre, seul leur emplacement change.
  3. Validez votre code avec le validator w3c en ligne ou depuis Brackets en utilisant l'extension « HTMLHint ».

### ETAPE 2 : Ecriture du code CSS pour desktop (1h30 – 2h00)

Nous allons « faire » de la dégradation responsive.

1. Créez votre feuille de styles. Faites le lien entre la page HTML et le fichier CSS.
  2. Intégrez la police (voir ressources fournies).
  3. Ecrivez le code CSS pour la gestion d'images fluides.
  4. Ecrivez le code CSS pour intégrer la mise en page « desktop ».
- Attention- remarques :**
- a. Donnez la priorité au positionnement des éléments ! Vous pourrez par la suite en fonction du temps restant, peaufiner.
  - b. Utilisez flexbox au maximum, cette méthode aide fortement !
  - c. Utilisez les unités relatives pour les largeurs de vos éléments et pour les tailles de polices pour la fluidité de la page. Pour plus de précision, vous pourrez prendre les dimensions depuis le logiciel de traitement d'images installé sur le poste.
  - d. Le slider (carroussel) : laissez-le de côté pour l'instant, remplacer le par une div contenant une simple image.
  - e. La flèche qui pointe vers le bas : placée en fin de première section, ne la gérez pas à cette étape, laissez pour plus tard.

## TP 1 – 2 : Dégradation responsive

---

### ETAPE 3 : Intégrer un menu hamburger (30min - 1h00)

La question 3 de cette étape nécessite une phase de recherche sur Internet.

1. Ecrivez la balise meta pour adapter votre viewport à la taille physique des supports mobiles.
2. En fin de votre fichier CSS, écrivez la media queries pour cibler les petites résolutions. Pour tester votre media queries, affichez une couleur de fond différente à la page lorsque cette plage de résolution est atteinte.
3. Intégrer un menu hamburger.

### ETAPE 4 : Media queries et CSS pour la mise en page des petites résolutions (tiny) (1h30)

1. Complétez votre requête media queries par le code CSS pour la mise en page des petites résolutions (*zoning\_tiny.xls*).

Rappels - Remarques :

- ✓ seules les différences entre les 2 mises en page doivent être traduites en CSS.
  - ✓ Lorsqu'un élément doit être modifié d'une mise en page à l'autre, réutilisez de préférence, le même sélecteur.
2. Adaptez les tailles des polices pour ces résolutions.
  3. Hébergez votre site et tester avec 2/3 téléphones différents de vos voisins. (Gestion des écrans haute résolution, Retina laissée pour plus tard).
  4. Testez sur tablette si possible.

### ETAPE 5 : Gestion des résolutions supplémentaires / Finalisation (1h30-2h30)

Pour obtenir une vraie page responsive web design, la page doit s'adapter quelle que la résolution. Il reste plusieurs autres cas à gérer.

1. Depuis votre navigateur, faites varier la largeur de la fenêtre. Repérez les résolutions à partir desquelles la mise en page « casse ». Améliorez en rajoutant d'éventuels media queries intermédiaires (police, positionnement, ...).
2. Observez votre site sur un téléphone haute résolution. Que remarquez-vous ? Proposez une solution.

----- Pour aller plus loin ...

3. Insérer un slider responsive (non déclenché automatiquement de préférence).
4. Sur la maquette desktop, à la fin de la section de bienvenue, la flèche pointant vers le bas avait été laissée de côté. Intégrez-la. Faites en sorte que cette section occupe la hauteur complète de l'écran desktop, quelle que soit la hauteur de l'écran.

## TP 1 – 2 : Dégradation responsive

## ANNEXE : maquette (desktop)

