## Infrastructure et Frameworks pour Applications Web

TP1 : HTML, CSS et JavaScript

MASTER 2 - IFAW Bekhouche Abdesslem/Sobral Diogo

Tout code doit être documenté.

## Exercice 1:

- 1. Créer un fichier contact.html permettant de rajouter un contact dans un annuaire. Le formulaire correspondant contiendra les champs suivants :
  - Name: First, middle et last name (text) dont la taille maximale vaut 10 et dont la valeur par défaut vaut "Nom";
  - Gender: male or female (bouton radio);
  - Phone, Email (text);
  - Contact (checkbox);
  - Et les deux boutons : Envoi pour soumettre le formulaire (dans un premier temps aucune action ne sera associée au bouton) et remise à zéro pour réinitialiser l'ensemble des champs.
- 2. Ajouter une image à votre page.

p).

- 3. Ajouter un paragraphe à votre page.
- 4. Créer un menu composé des éléments suivants :
  - a) « Menu local » renvoyant vers le paragraphe ;
  - b) « Passeport dauphine » renvoyant vers https://passeport.dauphine.fr/:
    - a. En ouvrant un nouvel onglet sur le navigateur ;
    - b. Accessible par le raccourci clavier « p » (ex : sur Chrome Alt +
- 5. Ajouter une feuille de style (fichier css) qui sera appliquée à l'ensemble de(s) page(s) précédemment créé(es).
  - La taille de la police doit être de 12pt ;
  - La font doit être « Arial »;
  - Les labels doivent être en gras :
  - Seul le label email doit être en « rouge ».
- 6. Ajouter un bouton « Appel javascript » qui se charge d'appeler une fonction javascript. Cette fonction javascript permettra de changer la valeur d'un élément html identifié par un id *pld* par la valeur spécifiée en paramètre *pValue*. L'appliquer au champ « Nom » avec une valeur « Name value changed ».

Paris Dauphine - Master IFAW TP 1 : Généralités Année 2024/2025

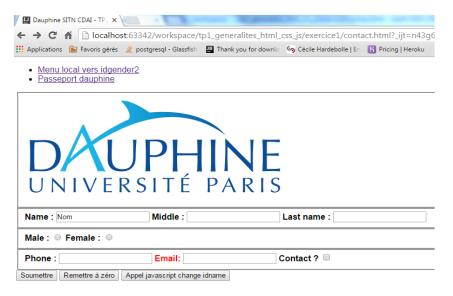


Figure 1: Illustration du besoin

- 7. Utiliser les outils mis à disposition par vos navigateurs pour :
  - a. Changer dynamiquement les styles css de votre page;
  - b. Utiliser l'ardoise javascript et manipuler les éléments du DOM;
  - c. Voir les échanges réseau au niveau du navigateur ;
  - d. Avoir une idée de la performance de chargement de la page et identifier les éléments les plus coûteux.
- 8. Refaire l'étape précédente avec un autre navigateur.
- 9. Valider votre (vos) page(s) HTML
  - a. Corrigez les éventuelles erreurs rencontrées.
- 10. Valider votre (vos) fichier(s) CSS
  - a. Corrigez les éventuelles erreurs rencontrées.

## Pour aller plus loin:

- Utiliser un gestionnaire de versionning de code (SCM Source Code Management) type git au travers un outil comme gitlab ou github par exemple ;
- Utiliser bootstrap pour gérer l'aspect responsive de votre site http://getbootstrap.com/ et testez-le sur différents affichages;
- Utiliser Angular (ou un autre framework JavaScript reactJS ou encore VueJS par exemple) pour étendre les possibilités du front (par exemple en manipulant le DOM de manière plus aisée) <a href="https://angular.io/">https://angular.io/</a>;
- Mettre en place un template de site pour votre future *potentielle* application.

## Liens utiles:

- <a href="http://www.w3schools.com/html/">http://www.w3schools.com/html/</a> pour l'ensemble des éléments HTML
- https://validator.w3.org/ validateur HTML
- https://jigsaw.w3.org/css-validator/ validateur CSS
- N'importe quel éditeur peut être utilisé, quelques-uns parmi les plus plébiscités :

- o <a href="https://www.jetbrains.com/webstorm/">https://www.jetbrains.com/webstorm/</a>
- o https://code.visualstudio.com/
  - Ou encore des éditeurs ne nécessitant pas d'installation et accessibles depuis un navigateur :
    - <a href="https://jsfiddle.net/">https://jsfiddle.net/</a>
    - https://codepen.io/
    - <a href="https://replit.com/">https://replit.com/</a>