

Modalité sur le rendu du mini-projet

Réalisation en Binôme ou monôme;

Remise par e-mail au plus tard le samedi 20/01/2024 ;

Présentation le dimanche 21/01/2024 ;

Remarque : toute copie de code sera sévèrement sanctionnée

Réalisation d'un guide de randonnées en ligne

L'objectif de ce travail est de créer un guide de randonnées en ligne. On souhaite avoir:

I. La page d'accueil

Elle fait office de page de recherche. Comme illustré ci-dessous :

The screenshot shows a search interface for a hiking guide website. The layout is as follows:

- Header:** Contains the title "Le titre de votre guide de randonnées".
- Nav (Left Column):** Displays student information: Nom : Toto, Prénom : Tutu, Spécialité : TI, Section : 1, Groupe : 2, Mail : t.toto@mail.dz. It also includes a link "[Ajouter Parcours](#)".
- Section (Main Content):** Contains fields for "Recherche": Ville (City), Commune (Town), Type (Type), Distance (Distance range slider), and a "Valider" (Validate) button.
- Resultat de la recherche (Bottom):** Shows search results: 1. [Mohamadia](#) (Alger) and 2. [Yakouren](#) (Tiziouzou). Each result has a checkbox and a small icon.

a) Présentation de la page

- La partie **header** doit contenir :
 - Comme titre le **nom de votre guide de randonnée** ;
 - Une **image de fond** de votre choix ;
- La partie **nav** doit contenir :
 - Les **informations des étudiants** qui ont réalisé le projet ;
 - Le lien [**Ajouter Parcours**](#) qui renvoie vers la page formulaire en mode création ;
- La section contiendra les champs de recherche **ville**, **commune**, **Type de randonnée**, **distance du parcours (format range)** et un bouton **valider** pour lancer la recherche ;
- La recherche peut être effectuée sur un ou plusieurs champs, saisie complète ou partielle.
Exemple : si on saisi dans le champ ville le mot « Al » le résultat doit être, toutes les randonnées dont le nom de la ville débute par « Al » ;

5. Le résultat de la recherche sera affiché en dessous des champs de recherche comme suit :
 - a. Liste contenant des parcours (résultat de la recherche), sous format de lien hypertexte ;
 - b. Icônes modification et suppression, à côté de chaque parcours affichée.

b) Fonctionnement de la page

1. Les informations affichées dans les parties **Header** et **Nav** doivent être chargées à partir d'un fichier XML nommé **Config.xml**.
 - a. Vous devez proposer un fichier **Config.xml** ainsi qu'un fichier nommé **Config.dtd**, pour la **DTD** qui va avec. Le fichier **Config.xml** doit contenir les informations suivantes :
 - Pour le **Header** : le **titre** et le **chemin** vers l'image de font
 - Pour le **Nav** : le **nom, prénom, spécialité, section, groupe** et **l'e-mail** des étudiants qui ont réalisé le projet.
 - b. Le fichier **Config.xml** doit être validé et parsé (lu) on utilisant l'**API DOM**.
2. Le lien **Ajouter Parcours**, renvoie vers la page formulaire (elle sera présentée plus loin) en mode création (formulaire vide) ;
3. La recherche doit être effectuée dans un fichier nommé **Parcours.xml** (**Parcours.xml** sera présenté plus loin) en fonction des champs de recherche renseignés.
 - a. Le fichier **Parcours.xml** doit être validé et parsé on utilisant l'**API DOM** et **XPath**.
4. Le résultat de la recherche sera affiché en dessous des champs de recherche comme suit :
 - a. Une liste contenant les noms des parcours sélectionnés, en forme de lien hypertexte avec les icônes modification et suppression;
 - b. Le lien renvoie vers la page du parcours concerné.
5. L'icône modification renvoie vers la page formulaire en mode modification, comme suit :
 - a. Les informations du parcours à modifier seront chargées à partir du fichier **ID.xml** concerné (**ID.xml** sera présenté plus loin) ;
 - b. Le fichier XML sélectionné, doit être validé et parsé on utilisant l'**API DOM** ;
6. L'icône suppression Permet de supprimer le parcours concerné dans le fichier XML **Parcours.xml**.

II. La page formulaire

Contient un formulaire pour la saisie des informations décrivant le parcours.

a) Présentation de la page

1. Le formulaire doit contenir les champs de saisis suivants :
Ville, Commune, Nom du Parcours, descriptif, Type du parcours, Distance, Niveau de difficulté, photos;
2. On rattache à chaque **parcours au moins** une **Photo**;
3. Le type du parcours prend les valeurs « **A pied, Vélo ou Cheval** »

4. Le niveau de difficulté prend les valeurs de **1 à 5**.
5. Dans le cas d'une modification, les champs du formulaire seront affichés avec les informations du parcours a modifiée.

NB : Tous les champs sont obligatoires et vous devez définir les tailles des champs.

b) Fonctionnement de la page

Les informations saisies dans le formulaire seront enregistrées dans deux fichiers **ID.xml** et **Parcours.xml** comme suit :

1. Un fichier **ID.xml** et créer à chaque ajout d'un nouveau parcours. Il doit contenir toutes les informations saisies;
 - a. **ID** est un ID unique qu'on affecte au parcours lors de sa création ;
Exemple : Le parcours dont l'ID est PR0001, son fichier sera **PR0001.xml** (**voir Annexe**)
2. Le fichier **Parcours.xml** (**Exemple en Annexe**) :
 - a. Servira à la recherche et contiendra les informations suivantes :
ville, commune, Nom, Distance et le Type du parcours
 - b. Il est créé lors de l'ajout du premier parcours;
 - c. Le fichier est enrichi à chaque ajout d'un nouveau parcours;
3. Dans le cas d'une modification on met à jour le fichier **ID.xml** concerné et le fichier **parcourss.xml**;
4. Vous devez proposer deux schémas XML, **ID.xsd** et **Parcours.xsd** qui définissent respectivement les modèles de documents **ID.xml** et **Parcours.xml**;
5. Les fichiers XML **ID.xml** et **Parcours.xml**, doivent être créés, validés, parsés et modifiés on utilisant l'**API DOM** et le **langage XPath**.

III. La page Parcours

Cette page affiche les informations d'un parcours.

a) Présentation de la page

Cette page affiche toutes les informations d'un parcours sélectionné. Vous organisez la page comme bon vous semble.

b) Fonctionnement de la page

Les informations affichées dans la page **Parcours.html** doivent provenir du fichier **ID.xml** concerné.

1. La page **Parcours.html** est le résultat de la transformation du fichier **ID.xml** en utilisant le **langage XSLT** et une **feuille de style Ville.xsl**
2. Vous devez proposer un seul fichier **parcours.xsl** qui permettra d'effectuer la transformation quelques soit le fichier XML **ID.xml**.

Important : Le Code doit être clair, propre et chaque instruction doit être commentée.

Annexe

***** Instance du fichier Parcours.xml *****

```
<?xml version="1.0" encoding="UTF-8"?>
<recherche>
    <villes>
        <ville nom="Alger">
            <parcours ID="PR0001" commune="Mohammadia">
                <nom>Mohamadia</nom>
                <distance unite="KM" long="1"/>
                <type>A pied</type>
            </parcours>
            <parcours ID="PR0002" commune="Bab Ezzouar">
                <nom>balade</nom>
                <distance unite="KM" long="3,4"/>
                <type>A pied</type>
            </parcours>
        </ville>
        <ville nom="TiziOuzou">
            <parcours ID="PR0005" commune="Idjer">
                <nom>Yakouren</nom>
                <distance unite="KM" long="5,9"/>
                <type>vélo</type>
            </parcours>
        </ville>
    </villes>
</recherche>
```

***** Fichier PR0001.xml *****

```
<?xml version="1.0" encoding="UTF-8"?>
<ville nom="Alger">
    <parcours ID="PR0001" commune="Mohammadia">
        <nom>Mohamadia</nom>
        <descriptif>marche autour d'un grand terrain nu au cœur de la capitale</descriptif>
        <distance unite="KM" longueur="1"/>
        <type>A pied</type>
        <niveau>2</niveau>
        <photos>
            <photo alt="lieu de départ" src="MohDépart.jpg"/>
            <photo alt="Djamaa el djazair" src="Djamaa.jpg"/>
        </photos>
    </parcours>
</ville>
```