

## **Adoption d'animaux de compagnie**

Vous devez concevoir une application web servant à offrir un animal de compagnie en adoption. L'adoption et la mise en adoption sont gratuites pour les utilisateurs.

### ***Consultation***

Les utilisateurs peuvent consulter la liste des animaux offerts pour une adoption. Chaque animal possède sa propre page affichant ses données et son adresse. La page d'accueil doit afficher 5 animaux disponibles au hasard ainsi que des liens vers leur page respective. Un moteur de recherche simple doit également permettre une recherche parmi les animaux disponibles.

Les résultats de la recherche doivent être affichés sur une nouvelle page. Les résultats doivent être affichés avec leur nom, leur description, ainsi qu'un lien vers la page de l'animal.

Sur la page d'un animal, un lien doit permettre de contacter le propriétaire de l'animal par courriel pour lui signifier notre intérêt pour l'adoption.

### ***Mise en adoption***

L'utilisateur peut mettre un animal en adoption en remplissant un formulaire contenant : le nom de l'animal, l'espèce de l'animal, sa race, son âge, une description textuelle, l'adresse courriel du propriétaire et l'adresse où récupérer l'animal.

Voici les validations que vous devez faire au niveau du frontend :

- Tous les champs du formulaire sont obligatoires;
- Aucun champ ne peut contenir une virgule.
- Le nom de l'animal doit avoir entre 3 et 20 caractères.
- L'âge doit être une valeur numérique entre 0 et 30.
- L'adresse courriel doit avoir un format valide.
- L'adresse doit comporter les champs suivants : adresse civique, ville, code postal. On assume que c'est toujours au Québec. Le code postal doit avoir un format canadien.

Si le formulaire n'est pas rempli adéquatement, les données ne doivent pas être soumises au backend et un message explicatif doit s'afficher à côté de chaque champ en erreur. Rappelez-vous que les validations doivent toujours être réalisées au niveau du frontend pour la convivialité et au niveau du backend pour l'intégrité des données.

### ***La base de données***

Une base de données SQLite vous sera fournie contenant des animaux en adoption. Les colonnes sont les suivantes :

- id : un identifiant unique pour l'animal;
- nom : le nom de l'animal;
- espece : le type d'animal (ex. Chien, chat, poisson, serpent);
- race : la race de l'animal (ex. Schnauzer);

- age : l'âge de l'animal en années;
- description : un texte court décrivant l'animal;
- courriel : l'adresse courriel du propriétaire de l'animal;
- adresse : l'adresse civique où récupérer l'animal lors de l'adoption;
- ville : la ville de l'animal;
- cp : le code postal.

Les données seront encodées en UTF-8.

Lors de la soumission du formulaire de mise en adoption d'un animal, votre backend doit ajouter l'animal à la base de données. L'identifiant est généré automatiquement par votre logiciel. Après la sauvegarde, l'utilisateur est redirigé vers la page de l'animal.

## **Messages**

Aucun message ne doit être affiché à l'utilisateur à l'aide des fonctions de popups de Javascript (ex. alert, prompt, etc.). Utilisez une autre technique impliquant du Javascript et la manipulation du DOM pour vos messages de confirmation et d'erreur. Toute erreur et confirmation doit être affichée à l'utilisateur.

## **Technologies**

Dans le front-end, vous **devez** utiliser les technologies suivantes :

- HTML 5
- CSS 3
- Javascript

Toujours dans le front-end, vous **pouvez** utiliser toutes les librairies et tous les frameworks Javascript que vous souhaitez, tant que ça ne nécessite aucune installation lors de la correction (vous fournissez les sources dans votre projet).

Dans le back-end, vous devez utiliser Python 3.13+ avec Flask 3+. Aucun module Python supplémentaire ne peut être utilisé.

Tous les fichiers sources doivent être encodés en UTF8.

Il est nécessaire d'utiliser le patron POST-REDIRECT-GET sauf si spécifié autrement.

Utilisez le squelette de projet qui est mis à votre disposition.

## **Remise**

Le travail peut être fait seul ou en équipe de 2 personnes. Le répertoire de travail contenant les fichiers doit être archivé dans un fichier zip et nommé selon le code permanent des auteurs.

L'archive doit être remise par Moodle. Aucun retard ne sera accepté et une pénalité sera appliquée pour une archive non conforme.

## **Pondération**

Respect des exigences : 50%  
 Convivialité du site web : 15%  
 Qualité du code : 15%  
 Aspect visuel du site : 20%