Documentation projet Web mapping MH04

**html :**

Rien de particulier à signaler à part qu’on retrouve les objets principaux affichés dans notre page à savoir :

* Un tableau de trois colonnes (Long, Lat, Adresse) qui initialement est vide. On le remplira par la suite dans le javascript.
* Un formulaire contenant deux boutons : page précédente et page suivante qui permet d’afficher de manière plus ergonomique le tableau et ainsi naviguer dans ce dernier.
* Une section « carte ». Ici, nous utilisons Leaflet pour afficher nos données.

**javascript :**

Tout d’abord nous commençons par initialiser notre map avec Leaflet en lui appliquant directement des valeurs par défauts comme la position initial et le niveau de zoom (.*setView()*). Nous reviendrons sur la variable *apiKey* par la suite.

Nous ajoutons aussi une couche marker qui va nous permettre d’ajouter sur la carte des pings affichants la position des pokemons sur la carte.

Ensuite nous définissons une fonction principale nommée « recup() » qui sera la fonction lancée à chaque nouvelle requête.

On commence par faire une requete server (donc vers le php), afin de récupérer dans notre base de donnée les coordonnées de nos pokemons.

Pour chaque ligne du résultat nous la mettons en forme, en commençant par séparer la longitude de la latitude qui sont collés dans notre base de donnée. Puis nous affichons un marker de style par défaut (bleu) sur la carte.

Ensuite nous construisons le tableau : j’ai commencé par choisir la taille du tableau que je voulais afficher, de manière à ce qu’il soit ergonomique. J’ai donc décidé d’afficher 5 résultats par 5. Pour cela, j’ai définit une variable « page » qui nous renseigne sur le numéro de la page actuellement affiché. Puis je parcours les 5 lignes précédents la page affichés. Nous aurions pu approfondir cette méthode pour augmenter l’efficacité de l’algorithme en faisant une requête server qui renvoie uniquement les 5 lignes voulus. Pour cela il aurait fallu envoyer dans le php la valeur de la page et faire une requete sql qui utilise cette valeurs pour ne renvoyer que 5 lignes à chaque fois. Mais par manque de temps je me suis concentré sur le modèle simple.

Pour ces derniers, j’ai décidé de changer le style des markers en les passant en rouge afin de voir lesquels sont actuellement affichés dans le tableau. En réalité on duplique les markers en question, mais sachant qu’on en ajoute uniquement 5, cela n’a que très peu d’impact sur les performances de nos machines. Ensuite, nous voulons trouver une adresse physique à partir des coordonnées. Ce processus s’appelle le *Reverse Geocoding*. Pour réaliser cela, nous allons utiliser le service proposé par ESRI sur Leaflet (.*reverseGeocode()*). Pour cela il faut d’abord récupérer une clé sur ESRI (payant), mais ayant un compte grâce à l’ENSG, nous y avons accès. Cette clé elle est ensuite stockée dans une variable ligne 5 (*apiKey*) et c’est au moment d’appeler la fonction nous renseignons cette dernière. Puis nous passons en entrée les coordonnées et le résultat recherché se récupère avec *result.adress.Match\_addr*.

Pour finir nous ajoutons une ligne par tableau avec des appendChild et nous rentrons pour chaque colonne les valeurs obtenus.

Enfin je défini les fonctions qui seront appelés avec les boutons (*addEventListener*), qui réinitialise le tableau à chaque clique, et qui incrémente ou décrémente la variable page.

**Php :**

On commence par se connecter à notre BDD commune à tous notre groupe.

Puis nous effectuons une requête sql qui de manière simple et naïve renvoie juste les 50 premiers résultats. Nous aurions pu approfondir en renvoyant par exemple les 50 résultats les plus proches de la position du joueur, ou encore introduire une barre de recherche permettant de rechercher tous les pokémons d’un certain type, etc.

Pour finir on formate le résultat sous forme de json en gardant le maximum d’information. Etant donné qu’au final dans la suite nous utilisons que les coordonnées peut etre aurait on pu regarder si on gagne vraiment en performance en gardant uniquement l’information souhaitée. Mais dans le doute j’ai gardé toutes les informations utiles.

Louis Steinmetz