

*Projet à faire en groupes de 3 ou 4 maximum.*

## **votre application web-MAP collaborative**

### **Présentation du projet**

Le but est de réaliser une application collaborative WEB full-stack, dont l'IHM contient pour partie un fond de carte interactif. Vous êtes libres du thème de votre application sous la condition de respecter les contraintes du projet, en particulier :

D'un point de vue technique

- des données du monde du libre (open data) doivent contribuer à définir le cœur de votre application et
- un site web applicatif associé à une base de données offrira des API en relation avec votre application.

D'un point de vue applicatif,

- l'application fonctionnera en mode connecté (l'utilisateur doit se connecter pour utiliser cette application)
- Elle organisera des éléments de coopération/collaboration pour un groupe d'utilisateurs autour de la carte. Il pourra y avoir plusieurs groupes.

Ce projet est à faire à 3 étudiants pris dans un même bi-groupe. Votre chargé de TD pourra autoriser 4 membres pour un projet mais avec 20% de travail en sus et un travail en 2 sous équipes. L'évaluation se fera en séance de TP lors de la dernière semaine de la période.

### **Spécifications fonctionnelles**

Le développement de la partie cliente de l'application sera fondé sur différentes capacités en liens avec une carte géospatiale, faisant référence à des coordonnées GPS (longitude, latitude) comme dans Google map, Leaflet ou Mapbox. On peut :

- placer des objets variés sur une carte à des positions géographiques type GPS (latitude-longitude).
- Animer ces objets et leur associer des informations physiques ou géospatiales (latitude-longitude).
- Réviser ces informations à tout moment en conservant la référence des objets ou motifs placés sur la carte.
- Définir des itinéraires, sous forme de points d'origine et de destination. Le calcul d'un itinéraire sur la carte peut être automatisé via des API existantes (notion de route ou routine) mais reste optionnel pour ce projet.
- Déplacer ou zoomer le fond de carte, comme afficher/recentrer la carte sur un point ou un objet, par exemple une des voitures.

L'application pourra fonctionner suivant 3 types de vues, accessibles au travers d'un système de menu ou d'onglets :

- un mode de connexion/deconnexion affichant le profil de l'utilisateur connecté
- un mode de définition et de configuration pouvant servir à la gestion collaborative d'un groupe où s'active l'utilisateur

- un mode applicatif pour l'activité principale de l'application.

L'affichage du fond de carte et l'interaction avec l'utilisateur seront développés sur la base de modules préconstruits comme :

- une bibliothèque de fonctions `javascript/jquery` prédéfinies, comme celles de `jquery-ui` pour la réalisation d'un interface utilisateur
- des bibliothèques de gestion de cartes comme celles de `Leaflet`
- des accès à des API distantes accessibles via des requêtes AJAX, comme ceux de `Nominatim` pour obtenir des positionnements géographiques en latitude et longitude, pays, ville, ...
- des fichiers `json` offrant des données variées en accès libre, qu'il faudra mettre à disposition comme des fichiers `json` de son site web. Alternativement, des API délivrant des données `json`.

## 1. Développement

La programmation de la partie cliente sera faite en `jQuery`. Il faudra aussi faire tourner un serveur web en back-end interfaçant les requêtes http nécessaire à la partie cliente, gérer une base de données `mysql`, écrire des scripts de services web en `php`. Le développement pourra se faire en 4 étapes :

### Etape 1

+ Prise en main du projet, découverte du sujet, découverte des outils logiciels développant la partie cliente de l'application, comme `leaflet`, en association avec les serveurs `openstreetmap` et `nominatim` pour la gestion des coordonnées et des adresses.

+ constitution ou exploitation de fichiers des données libres sur Paris (voir par exemple open data de Paris, de velib, qui sont en accès libre sur le WEB). Exploitation d'APIs complémentaires libres et abonnement préalable pour celles d'entre elles le nécessitent.

### Etape 2

+ Préparation/affichage de la carte.

+ Lecture des données, constitution et affichage de marqueurs et groupes de marqueurs (layers)..

### Etape 3

+ Développement de la partie cliente sur le navigateur, comme `Firefox` ou `Chrome`. Mise en œuvre des interactions avec l'utilisateur, sur la carte, les marqueurs et formes dessinées.

### Etape 4

+ Développement de la partie serveur de votre application. Préparation de la base de données, écriture des services web en interaction avec cette base

### 3. Evaluation du projet

Nous porterons une attention aux points technologiques suivants :

- + ergonomie de votre application
- + intérêt collaboratif de votre application
- + richesse des données exploitées, json et API publiques
- + affichage de cartes et de listes d'objets graphiques sur la carte en affichage multi-couches
- + modification dynamique de la carte, de ses objets ou des informations associées (approche événementielle avec exploitation d'informations géospatiales.
- + gestion d'une base de données applicative sortant du cas d'école (avec au moins 3 tables) et
- + services back-end épaulant votre application.

Le projet inclura un **rapport** mettant en valeur votre cahier des charges, les langages et points de développement significatifs, une description de l'application et des écrans commentés, une description de votre base (diagramme commenté) et des services back-end mis en œuvre, les évolutions à prévoir.

Le développement logiciel de votre application devra être équilibré entre les membres du votre projet, sous peine d'une baisse globale à la notation.

Vous pouvez prendre certaines libertés avec le sujet, sous l'autorisation explicite de votre enseignant - chargé de TD.

## 4. Annexes et rappels : aspects technologiques pour la gestion des cartes

L'objet Leaflet (noté L) offre par ses fonctions un vrai langage de gestion de cartes permettant de charger des cartes de façon compositionnelle (cartes composées de tuiles – morceaux graphiques de la carte précompilés), en prenant en compte le niveau de zoom désiré, des notions d'adresses et de coordonnées géospatiales (longitude-latitude). Il s'appuie sur des serveurs de cartes et de données géospatiales.

Le site de Leaflet a un onglet `quickstart`, mais aussi de petites applications démo et aussi la documentation du langage (i.e. ses fonctions). Les fonctionnalités de Leaflet sont décrites sous l'onglet `doc`. Il est indispensable de comprendre comment définir et gérer : (1) des `markers`, leur apparence et les informations associées s'affichant dans une fenêtre `popup`, ou des formes graphiques plus spécialisées comme des cercles rectangles ou polygones (2) des couches se sur-ajoutant à la carte, dites `layers` pour représenter un (sous) ensemble des `markers`, (3) des gestionnaires d'événements sur la carte, sur les `markers`, ou les autres formes graphiques, sachant que ces événements sont implicitement étendus pour fournir une information géospatiale au moment de l'événement.

Pour leaflet, un exemple d'utilisation de cette librairie est donné à l'adresse suivante sur le site de leaflet:

<http://leafletjs.com/examples/quick-start.html>

<https://leafletjs.com/examples.html>

La première étape est d'importer l'objet Leaflet, nommé L, depuis votre page web. Vous pourrez ensuite utiliser les fonctions définies par cet objet, défini sur la base de jquery.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
```

```
<script src="http://cdn.leafletjs.com/leaflet-0.7.1/leaflet.js"></script>
```

On peut ensuite regarder la documentation sur le site et des tutos sur internet suivant les besoins.

<https://leafletjs.com/reference.html>

### Exemple de chargement d'un fond de carte

Le webservice principal que vous interrogerez pour récupérer les informations géographiques est <http://nominatim.openstreetmap.org>. Nominatim (du Latin, 'par le nom') est un webservice permettant de chercher des données Open Street Map (OSM) par leur nom et leur adresse (Exemple : recherche pays France). Les différents paramètres sont disponibles à l'adresse suivante

<http://wiki.openstreetmap.org/wiki/FR:Nominatim>

Exemple d'utilisation du service pour récupérer les coordonnées géographiques (latitude, longitude) de la France :

<https://nominatim.openstreetmap.org/search?country=france&city=paris&limit=1&format=json&addressdetails=1>

Exemple de résultat retourné au client au format JSON :

```
[{"place_id":"9157173744",
```

```
"licence":"Data \u00a9 OpenStreetMap contributors, ODbL 1.0.http://www.openstreetmap.org/copyright",
"osm_type":"relation","osm_id":"2202162",
"boundingbox":["-85.0500030517578","51.2683181762695","-178.38737487793","172.305725097656"],
"lat":"46.603354","lon":"1.8883335",
"display_name":"France"]}}
```

En rajoutant city=paris dans la requête, on obtient :

```
[{"place_id":297417241,"licence":"Data © OpenStreetMap contributors, ODbL 1.0.
https://osm.org/copyright","osm_type":"relation","osm_id":7444,"boundingbox":["
48.8155755","48.902156","2.224122","2.4697602"],"lat":"48.8588897","lon":"2.320
0410217200766","display_name":"Paris, Île-de-France, France
métropolitaine,France","class":"boundary","type":"administrative","importance":
1.0317101715588672,"icon":"https://nominatim.openstreetmap.org/ui/mapicons/poi_
boundary_administrative.p.20.png","address":{"suburb":"Paris","city_district":"
Paris","city":"Paris","ISO3166-2-lvl6":"FR-75","state":"Île-de-
France","ISO3166-2-lvl4":"FR-IDF","region":"France
métropolitaine","country":"France","country_code":"fr"}}}]
```

Une autre possibilité offerte par Nominatim est la fonction 'reverse' qui permet de retrouver des informations géographiques (ville, pays) à partir de coordonnées géographiques (latitude, longitude).

Exemple d'utilisation de la fonction reverse (pour un point (lat,long) situé en Angleterre) :

<http://nominatim.openstreetmap.org/reverse?format=json&lat=52.5487429714954&lon=-1.81602098644987&limit=1&zoom=4&addressdetails=1>

Exemple de résultat (retourné au client au format JSON) :

```
{"place_id":"97282697","licence":"Data \u00a9 OpenStreetMap contributors, ODbL 1.0.  
http://www.openstreetmap.org/copyright","osm_type":"relation","osm_id":"62149","lat":"54.7023545","lon":"-  
3.2765753",  
"display_name":"Royaume-Uni",  
"address":{"country":"Royaume-Uni","country_code":"gb"}}
```

Vous pourrez tester le rendu de vos requêtes d'adresse graphiquement sur le site :

<https://www.openstreetmap.org/search?query=paris#map=12/48.8589/2.3469>

Ici, 12/48.8589/2.3469 demande 12 comme niveau de zoom, suivi d'une coordonnée géospatiale.

Voici un lien pour un aperçu des attributs requetes openstreetmap et notamment de leurs attributs :

<https://www.geoapify.com/openstreetmap-geocoding>

## Exemple d'utilisation de AJAX et Json pour récupérer des informations localisées

La récupération des données géographiques sur le serveur se fera par le biais de requêtes AJAX qui permettront de récupérer dynamiquement dans la base de données les informations utiles à la personnalisation de la carte. Exemple de requête AJAX :

```
$.ajax({  
    type: 'GET',  
    url: "http://nominatim.openstreetmap.org/search",  
    dataType: 'json',  
    jsonpCallback: 'data',  
    data: {format: "json", limit: 1, country: "france", json_callback: "data" },  
    error: function (xhr, status, error) {  
        alert("ERROR "+ error);  
    },  
    success: function(data){  
        //La donnée Json reçue a été allouée en mémoire et est représentée par data  
        //...  
    }  
});
```

Cette requête permet de récupérer des informations géographiques (latitude, longitude, cadre-rectangle dit bounding box, ici de la France.

## Exemples de chargement d'un fond de carte routière

Exemple pour créer un fond de carte en JS (en supposant qu'un bloc DIV avec l'identifiant "map" existe dans la page html) :

```
//Chargement initial de la MAP dans le bloc DIV d'identifiant mappy
var map = L.map('map').setView([14,-14.8883335],4);
//ajout du fond de carte
L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {attribution:
'PING'}).addTo(map);
```

## Gestion d'événements augmentés munis d'informations géo-spatiales

```
map.on('click', function(e) { //e décrit l'événement augmenté
    alert(e.latlng); //voir l'objet Latlng
    console.log('latitude : ' + e.latlng.lat) ;
    console.log('longitude : ' + e.latlng.lng) ;
} );
```

Ou

```
function onClick(e) { ...
}
map.on('click', onClick); //référence à la fonction onClick()
map.off('click', onClick); //qu'il faut écrire.
```

**Note :** ce n'est pas au développeur d'instancier le paramètre <e> et ses attributs d'objet. C'est au thread du navigateur devant s'occuper de répondre aux clics sur la carte d'invoquer la fonction onClick(e) sur un clic et donc de définir « e ».

## Gestion des marqueurs et des layers de marqueurs

Les marqueurs ont des options tel que déplaçable (draggable), ils ont une image, un zone d'information qui leur est attaché de type popup. Le contenu du popup peut être un texte, un code HTML, une image ..., si vous voulez que le popup s'affiche au chargement de la page, vous pouvez faire ainsi ajouter la fonction openPopup().

```
var m = L.marker([2,48], {
    Icon :logo,
    draggable :true,
    opacity :0.8
});
m.addTo(map)
    .bindPopup('<h3>joli point d'intérêt</h3>')
    .openPopup();
```

### Définition d'un écouteur de clic sur un marqueur

```
function markerOnClick(e) {
    alert("Marker clicked");
}
```

```
Var m= L.marker([45.805073, 4.765305]).addTo(map);  
m.on('click', markerOnClick) ;
```

### **Comment spécifier des layers regroupant des familles de markers**

<https://www.mystranding.fr/leaflet-grouper-les-elements>

### **Comment distinguer des marqueurs en leur attribuant un ID**

Il est possible de les définir tous directement à partir d'une collection de données en json. La fonction L.geoJSON est exploitée (faisant référence à un flux représentant un ensemble d'objets geoJson).

<https://codetheworld.net/2020/06/leaflet-identifier-et-personnaliser-un-marker-avec-geojson/>

<https://leafletjs.com/reference.html#geojson>

<https://stackoverflow.com/questions/25683871/assign-id-to-marker-in-leaflet#answer-31032786>