Mélodie Fleury - Ing2

**PROJET WEBMAPPING**

Le but de ce projet est la création d’une carte interactive des fromages AOC de France pour « L’ami Molette », une association d’amateurs de fromages. Il souhaite une carte avec deux menus déroulants : un pour la sélection de la région fromagère et un pour la sélection des fromages AOC. Lorsqu’une région est choisie, seul les fromages AOC lui appartenant apparaissent. Lorsqu’un fromage AOC est sélectionné, alors les compteurs sont remis à zéro et seul l’AOC concernée est visible sur la carte. Ces données seront sous la forme de polygones. Enfin, lorsque l’utilisateur clique sur les marqueurs correspondants aux AOC, une fiche descriptive apparaît qui communique des informations plus fournies sur ce fromage.

Pour ce faire, nous avons accès à deux tables : « AOC\_fromages.csv », « comagri-communes-aires-ao.csv » et d’un fichier géojson des communes de France trouvé sur le site internet de l’insee. Dans la première table, il y a des informations à propos des Appellations d’Origines Contrôlées (AOC) des fromages : son nom, l’origine animale du lait, son type, son aire de production et enfin l’année d’obtention de cet AOC. Dans la seconde table, nous disposons des informations liées à la localisation des AOC avec l’attribut « Aire de production ».

**13/12/2021 :** Début du projet

* Découverte du sujet
* Discussion en visio-conférence avec Nicolas pour comprendre plus en détail les attendus du projet
* Apparition de la carte à l’aide des fonds de carte Leaflet

**14/12/2021 :** Absente finale océan hackathon à Brest

**17/12/2021 :** Création de la base de données. En effet, pour pouvoir faire la carte avec les marqueurs correspondants aux fromages. Il me faut ma propre base de données. Il est nécessaire de lier les deux tables à l’aide des attributs « Appellations d’Origines Protégées » (pour la table des AOC) et « Aire géographique » (pour la table de la localisation). La nouvelle table sera appelée « AOC\_fromages ».

Voici le code SQL utilisé :

SELECT \*

FROM "AOC\_fromages" as a

INNER JOIN comagri\_communes as c ON a."AppellationDorigineControlee" = c."Aire geographique";

**11/01/2021 :** Ma base sera constituée des attributs :

* « Aire de production » (Départements),
* « AOC » (nom),
* « IDA » (code de l’AOC),
* « Commune » (nom d’une commune appartenant à l’aire de production où se produit l’AOC),
* « CI » (code de la commune),
* « Long » (longitude de la commune),
* « Lat » (latitude de la commune),

Grâce à ces coordonnées, il sera possible de placer les marqueurs des fromages

* « Lait » (Provenance animale du lait),
* « Type » (Type du fromage produit),
* « Année obtention AOC »,
* « Autre nom géographique » (du lieu de l’AOC)

**15/01/2021**

Maintenant, pour faciliter le traitement des données, nous nous concentrerons sur le département de la Haute-Savoie. Ainsi, il est nécessaire de filtrer la base de données « AOC\_fromages » pour ne prendre que les fromages AOC appartenant à ce département et ses communes associées. Cette nouvelle table se nommera « AOC\_HauteSavoie ».

**22/01/2021**

Pour associer les polygones aux communes, il est nécessaire d’importer ce fichier géojson dans le serveur postgresql. Pour ce faire, nous utilisons le projet OsGeo4W qui contient une librairie libre appelée GDAL. Ce dernier permet de convertir des données d’un format de fichier à un autre en utilisant l’outil « ogr2ogr ».

Dans la console de OSGEO4W, on écrit :

*ogr2ogr -f "PostgreSQL" PG:"host=localhost port=5432 dbname=FromageAOC user=postgres password=postgres" D:\les\_documents\ProyectoWebMapping\Codes\_source\documentosIniciales\COMMUNES\_GEOJSON.geojson -nln Communes\_Geojson -a\_srs EPSG:4326*

Cela va donc transformer le fichier GeoJson en format PostGreSQL. Bien faire attention à avoir installé la cartouche spatial PostGis.

Pour cela, il suffit d’écrire comme requête SQL :

*-- Enable PostGIS (as of 3.0 contains just geometry/geography)*

*CREATE EXTENSION postgis;*

*-- enable raster support (for 3+)*

*CREATE EXTENSION postgis\_raster;*

*-- Enable Topology*

*CREATE EXTENSION postgis\_topology;*

*-- Enable PostGIS Advanced 3D*

*-- and other geoprocessing algorithms*

*-- sfcgal not available with all distributions*

*CREATE EXTENSION postgis\_sfcgal;*

*-- fuzzy matching needed for Tiger*

*CREATE EXTENSION fuzzystrmatch;*

*-- rule based standardizer*

*CREATE EXTENSION address\_standardizer;*

*-- example rule data set*

*CREATE EXTENSION address\_standardizer\_data\_us;*

*-- Enable US Tiger Geocoder*

*CREATE EXTENSION postgis\_tiger\_geocoder;*

**23/01/22**

Maintenant, il est nécessaire de ne prendre les géométries seulement des communes appartenant à la Haute-Savoie :

SELECT \*

FROM "communes\_geojson" as c

WHERE c."insee" LIKE '74%';

Cette nouvelle table s’appelle “Communes\_geojson\_haute\_savoie ».

Une fois cette étape effectuée, il faut réaliser une jointure entre les tables « AOC\_HauteSavoie » et « Communes\_geojson\_haute\_savoie » :

SELECT \*

FROM "AOC\_HauteSavoie" as a

INNER JOIN "Communes\_geojson\_haute\_savoie" as c ON a."Commune" = c."nom";

Nous avons alors notre table finale avec les données qui seront utilisées pour la carte.

#### Vérification connexion postgresql et mamp : PostgreSQL[#](https://ensg_dei.gitlab.io/web-az/principes/langages-serveur/" \l "postgresql" \o "Permanent link)

Pour se connecter à une base postgres en PHP, il faut d’abord le configurer :

* dans le fichier php.ini, décommentez les lignes (enlever le point-virgule) chargeant les extensions, notamment extension=php\_pgsql.dll
* dans le fichier httpd.conf, chargez la librairie dynamique avec LoadFile "C:/MAMP/bin/php/php[version]/libpq.dll" (exemple avec MAMP, vérifiez bien le dossier de votre version de PHP)
* redémarrez le serveur Apache