GHG result : Stage AKUO ENERGY

Compétences :

- **Gérer le patrimoine informatique.**
- **Répondre aux incidents et aux demandes d'assistance et d'évolution**
- **Mettre à disposition des utilisateurs un service informatique**
- **Travailler en mode projet**

AKUO ENERGY en tant qu'entreprise œuvrant dans le secteur des énergies renouvelables assure la connexion entre toutes les plateformes (sirh, crm, etc.) et centralise les informations de manière efficace grâce à un logiciel développé en interne . Ce système, connu sous le nom d'AKDB3 se sépare en 3 parties.

La 1ère partie, notamment la première à avoir exister est la base de donnée développé sur DBeaver, c'est le référentiel de données de l'entreprise.

La 2ème partie est le back, le back ou (mdm) a été développé avec la base de données afin de créer une jonction entre les différentes plateformes (SIRH, CRM…). C'est la partie la plus importante.

La 3ème partie est le front, donc simplement le site web, développé à partir du back, il permet de rendre lisible les données facilement et efficacement par tous les membres de l'entreprise.

On part d'une base de donnée qui a ensuite eu un back puis un front.

Description de la mission confiée :

Lors de ce stage, j'aurai pour mission la finalisation et l'optimisation d'une partie de la version 2 de AKDB afin que cette dernière puisse exister dans la version finale de AKDB (AKDB3).

Voici la page à modifier :

(page GHG AKDB2)

La page a rajouté se nomme ghg, elle regroupe les données solaire, hydraulique et éolienne classée selon l'année, le pays ou l'énergie demandée. Les données apparaissent sur la carte lorsque le curseur passe sur le pays en question ainsi que dans un encadré "résultats".

Il y a aussi une gradation de couleur prenant en compte le taux de CO2 par MWH, la couleur noire indiquant donc un taux élevé et un zoom sur les continents.

Le **GHG** Protocol fournit des guides sectoriels pour permettre de réduire le temps et le coût requis pour élaborer des systèmes de comptabilisation et de déclaration des GES pour les activités spécifiques des organismes.

Application :

Pour l'élaboration de ce formulaire, j'utiliserai le framework de angular, PHP, scss, HTML et typescript sur l'IDE PHPstorm.

Nous utilisons notamment wamp server et apache ainsi que DBeaver (base de données).

Afin de commencer, je dois établir les routes nécessaires, cela permet de définir quel composant afficher en fonction de la route.

Il faut d'abord se situer dans le dossier se chargeant du back, celui-ci s'appelle donc API, on se dirige ensuite logiquement dans le dossier router, ou l'on crée un fichier php, ici : "ghg.php".

Voici le code :

```
<?php    class GhgRoutes
{       protected static object *$routes*;
public static function getRoutes(): object
{         self::*$routes* = Router::*addSegment*("ghg",
[              Router::*addVerb*(HttpVerbs::*get*,
Router::*addRoute*(function() {
`require_once(*__DIR__* . "/../../../app/ghg/read/index.php");
              })              ),
],
[],
RoutesRights::*setRights*(null, null, null, null, true)          );
return self::*$routes*;       }`
```

On crée une classe ghgroute, on assigne alors des méthodes, notamment une pour établir des droits ou encore pour HTTP.

On vient ensuite lui expliquer ou il trouvera le code à exécuter avec :

```
require_once(*__DIR__* . "/../../../app/ghg/read/index.php");
```

ceci permet de dire que la requete que l'on veut exécuter se trouve dasn le dossier app, ghg, read puis index.php.

On vient ensuite placer le chemin de la route dans le dossier des routesGrouper comme ceci :

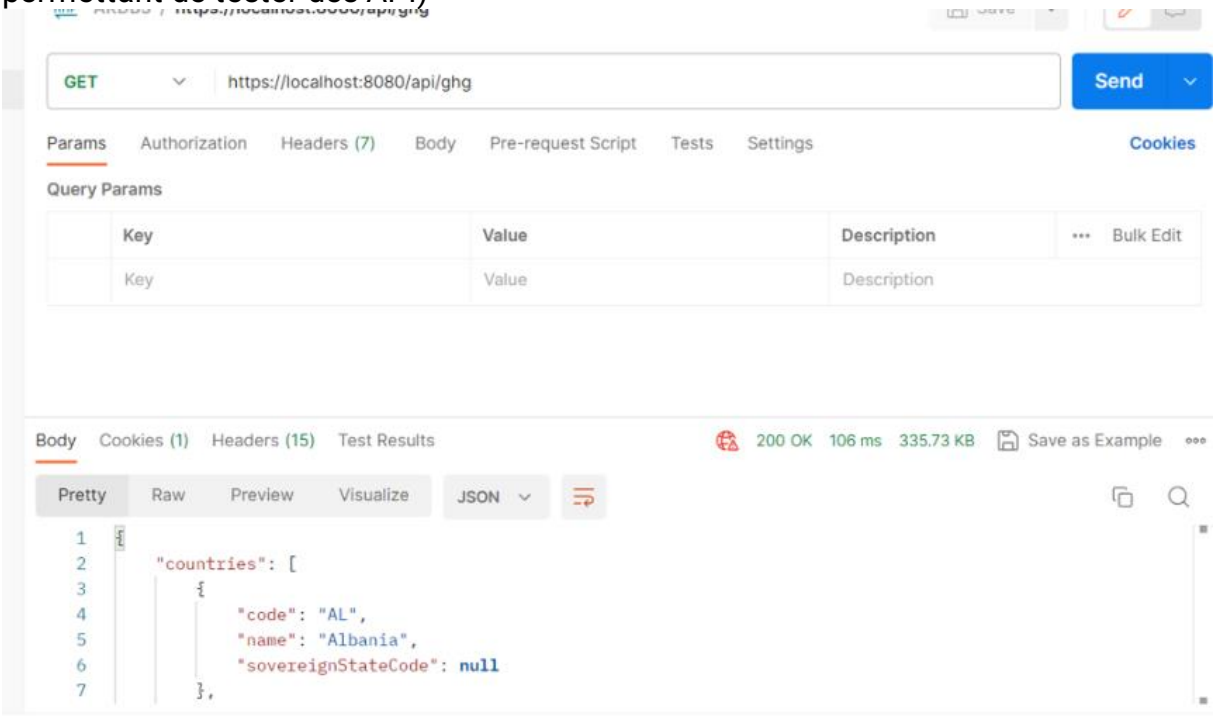
```
require_once(*__DIR__* . "/./ghg/routes.php");
```

Puis :

```
GhgRoutes::*getRoutes*(),
```

Seulement après cela, le système pourra comprendre quoi effectuer et où le chercher.

Nous pouvons ensuite venir tester la requête avec Postman(une application permettant de tester des API)



Lorsque mes routes fonctionnent, je commence à m'occuper du front, je crée les fichiers nécessaires dans front/app/src je crée le dossier ghg qui sera composé de ghg.component.ts, ghg.module.ts, ghg.component.scss, ghg.component.html.

Lorsque mes fichiers sont crées et remplis a l'aide des chemins et des fichiers que j'aurait indiqué dans les fichiers ts, je dois intégrer ma route au front dans le dossier général des routes, je rentre alors selon les exemples :

{ path: 'ghg', component: GhgComponent, canActivate: [MsalGuard], title: 'GHG', loadChildren: () => import('./ghg/ghg.module').then((m) => m.GhgModule)},

Etant donné que les routes sont intégrées et les liens bien claires pour l'appli, tout s'est lié ce qui permet des raccourcis.

Tout est en place afin que je commence a créer le formulaire GHG.

Pour commencer, je m'occupe du HTML dans lequel je n'ai qu'à importé des templates qui feront la mise en page du site selon l'identité de akuo.

Je vais donc prendre exemple sur les fichiers similaires de akdb3 et importé des templates, des fonctions et des classes pour faire apparaître la carte, le header, la jauge ,etc.

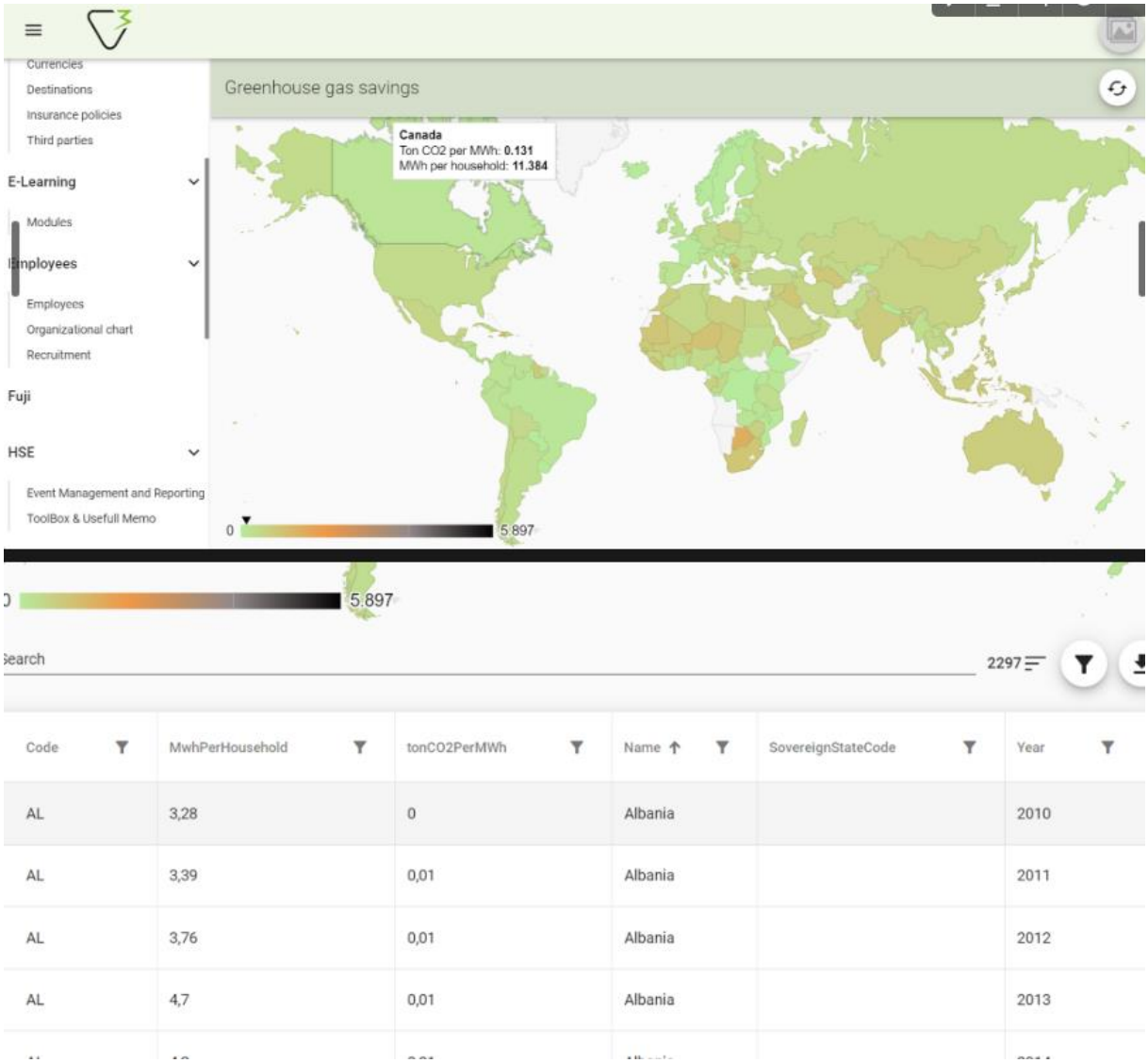
Voici mon premier rendu, où, afin de m'entraîner je crée ce tableau regroupant les données de la carte.

Pour cela, j'inclus un template tableau dans le HTML, crée une fonction et ajoute les imports nécessaires.

Voici le code qui permet de pousser les données dans le tableau :

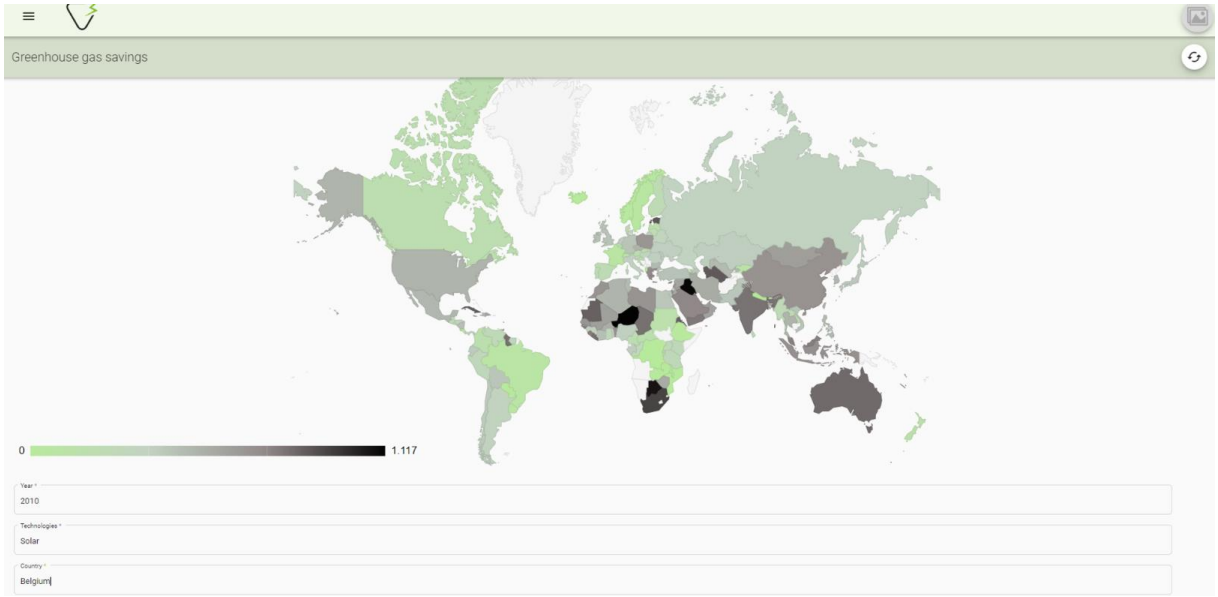
```
setMapData(form: any) {
// Map data
// Reset map data  this.mapData = []**;**
// Push map data  this.def.tableDef.data.forEach(country => {
// Push map data if year equals selected year
if (Number(country.year) === Number(form.year)) {
this.mapData.push([
country.code**,**
country.name**,**
Number(country.tonCO2PerMWh)**,**
Number(country.mwhPerHousehold)
])**;**
}
})**;**
// Force map refresh
this.mapData = [...this.mapData]**;**}
```

Voici le rendu :



J'enlève donc ce tableau et modifie les couleurs afin que cela soit plus agréable, puis ajoute des listes déroulantes pour les années, les énergies, et les pays à l'aide de template et de fonction/classe déjà créer.

Voilà le résultat :


Greenhouse gas savings
0
1.117
Year *
2010
Technologies *
Solar
Country *
Belgium