

## Mission réalisée E4 : Adaptation d'un bundle Symfony sur un projet (passage de highcharts à chartjs).

Centre de formation :					Option :	<input type="checkbox"/> SISR	<input type="checkbox"/> SLAM
<div>Compétences mises en œuvre</div> <div>Réalisations professionnelles (intitulé et liste des documents et productions associés)</div>	Période (sous la forme du JJ/MM/AA au JJ/MM/AA)	Gérer le patrimoine informatique	Répondre aux incidents et aux demandes d'assistance et d'évolution	Développer la présence en ligne de l'organisation	Travailler en mode projet	Mettre à disposition des utilisateurs un service informatique	Organiser son développement professionnel
		<ul style="list-style-type: none"><li>Recenser et identifier les ressources numériques</li><li>Exploiter des référentiels, normes et standards adoptés par le prestataire informatique</li><li>Mettre en place et vérifier les niveaux d'habilitation associés à un service</li><li>Vérifier les conditions de la continuité d'un service informatique</li><li>Gérer des sauvegardes</li><li>Vérifier le respect des règles d'utilisation des ressources numériques</li></ul>	<ul style="list-style-type: none"><li>Collecter, suivre et orienter des demandes</li><li>Traiter des demandes concernant les services réseau et système, applicatifs</li><li>Traiter des demandes concernant les applications</li></ul>	<ul style="list-style-type: none"><li>Participer à la valorisation de l'image de l'organisation sur les médias numériques en tenant compte du cadre juridique et des enjeux économiques</li><li>Référencer les services en ligne de l'organisation et mesurer leur visibilité.</li><li>Participer à l'évolution d'un site Web exploitant les données de l'organisation.</li></ul>	<ul style="list-style-type: none"><li>Analyser les objectifs et les modalités d'organisation d'un projet</li><li>Planifier les activités</li><li>Évaluer les indicateurs de suivi d'un projet et analyser les écarts</li></ul>	<ul style="list-style-type: none"><li>Réaliser les tests d'intégration et d'acceptation d'un service</li><li>Déployer un service</li><li>Accompagner les utilisateurs dans la mise en place d'un service</li></ul>	<ul style="list-style-type: none"><li>Mettre en place son environnement d'apprentissage personnel</li><li>Mettre en œuvre des outils et stratégies de veille informationnelle</li><li>Gérer son identité professionnelle</li><li>Développer son projet professionnel</li></ul>
Réalisations en milieu professionnel en cours de seconde année							
Adaptation d'un bundle Symfony (passage de highcharts à chartjs),		24/01/2022 au 31/01/2022					

Lors de ce second stage, la mission que l'on m'a demandé de réaliser est :

- Le remplacement d'un bundle symfony (de passer du bundle highcharts à chartjs).

**Sur cette documentation, je vais généraliser comment j'ai répondu à cette mission.**

Afin de pouvoir répondre à cette mission, je me suis tout d'abord renseigné sur ce qu'est Hightcharts et Chartjs.

Hightcharts et Chartjs sont des librairies symfony qui permettent la création de graphique interactifs et dynamiques en Javascript.

Avec l'arrivée en 2020 de Symfony UX qui est un nouvel écosystème au tour de Javascript pour le framework Symfony. Permettant de faire des créations dynamiques à l'aide du langage Javascript sans avoir besoin de le comprendre. Celui-ci utilise la librairie Chartjs, c'est pour cette raison que nous avons changé de librairie.

J'ai appris à utiliser Chartjs en suivant la documentation Symfony UX :

<https://symfony.com/blog/new-in-symfony-the-ux-initiative-a-new-javascript-ecosystem-for-symfony>

Et en utilisant ce tuto youtube :

[https://www.youtube.com/watch?v=ZuWFaNx-ZU0&t=1026s&ab\\_channel=YoanDev](https://www.youtube.com/watch?v=ZuWFaNx-ZU0&t=1026s&ab_channel=YoanDev)

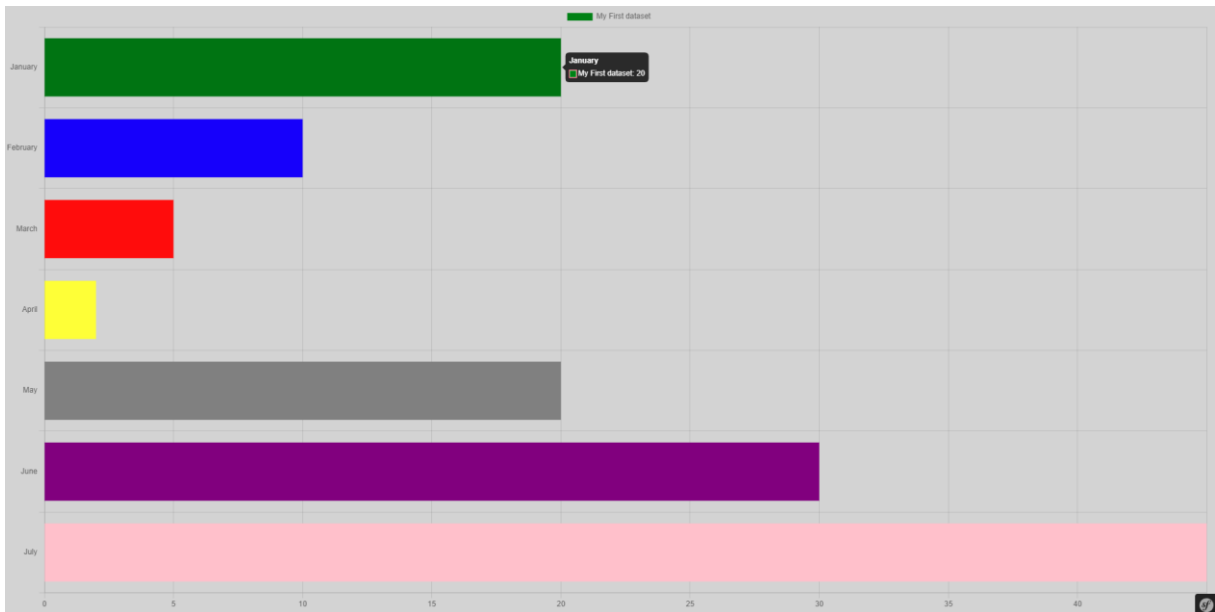
Pour installer Chartjs il faut faire un composer require :

```
- composer require symfony/ux-chartjs
```

Lorsque nous avons installé chartjs celui-ci va créer une classe ChartjsController automatiquement. C'est dans celle-ci que nous créons les graphiques.

Afin d'expliquer comment faire pour créer des graphiques avec ChartJs j'ai créé un graphique en guise d'exemple.

Le rendu du graphique :



Le code du graphique :

```
public function index(ChartBuilderInterface $chartBuilder): Response
{
    $chart = $chartBuilder->createChart( type: Chart::TYPE_BAR);
    $chart->setData([
        'labels' => ['January', 'February', 'March', 'April', 'May', 'June', 'July'],
        'datasets' => [
            [
                'label' => 'My First dataset',
                'backgroundColor' => ['green', 'blue', 'red', 'yellow', 'grey', 'purple', 'pink'],
                'borderColor' => 'rgb(255, 99, 132)',
                'data' => [20, 10, 5, 2, 20, 30, 45],
            ],
        ],
    ]);

    $chart->setOptions([
        'indexAxis' => 'y',
        'scales' => [
            'y' => [
                'suggestedMin' => 0,
                'suggestedMax' => 45,
            ],
        ],
    ]);

    return $this->render( view: 'chartjs/index.html.twig', [
        'chart' => $chart,
    ]);
}
```

Donc afin de créer un graphique, il faut tout d'abord renseigner la variable qui stockera le graphique. Ensuite il faudra utiliser `chartbuilder->createchart` permettant de créer le graphique du type renseigné.

Puis pour gérer les données du graphique, il y'a la fonction `setData`.

Dans celle-ci nous pouvons renseigner les labels qui permettent de nommer les données insérées dans le graphique, dans notre cas lorsque nous passons la souris sur la colonne verte cela nous affichera January, la bleue sera March etc...

Ensuite nous devons renseigner les données dans datasets dans celle-ci le label sera utilisé pour nommer le graphique. Le `backgroundcolor` et `borderColor` permettront de coloriser les colonnes. Et le data d'insérer des valeurs.

Nous avons aussi une fonction permettant d'ajouter d'autres options qui est la fonction `setOptions`, ici dans celle-ci j'ai indiqué sur quel axe est basé le graphique mais aussi j'ai rajouté un système d'échelle permettant de mieux représenter les valeurs.

Et à la fin nous retournons la vue afin d'afficher le graphique.

Sur la vue :

```
{% extends 'base.html.twig' %}

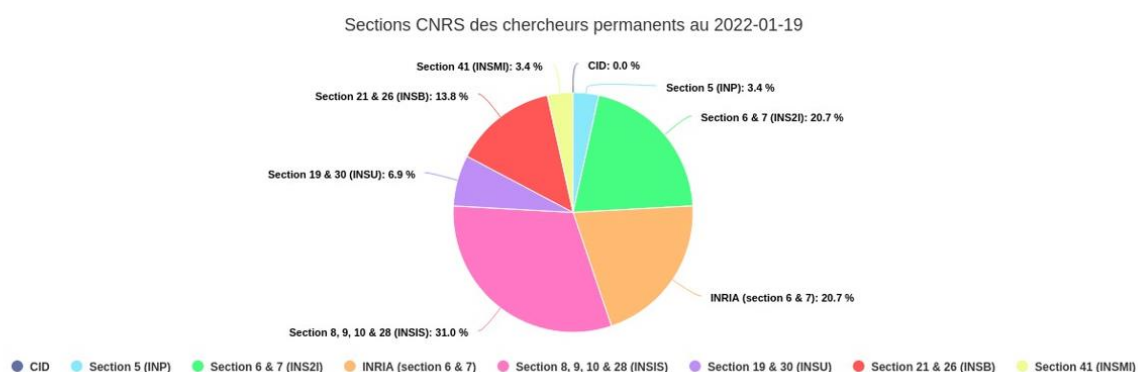
{% block body %}

<div class="example-wrapper">
    {{ render_chart(chart) }}
</div>
{% endblock %}
```

Afin d'afficher le graphique sur la vue, il faut rajouter `render_chart`(variable du graphique).

C'est globalement de cette façon que j'ai procédé afin de recréer les graphiques en chartjs.

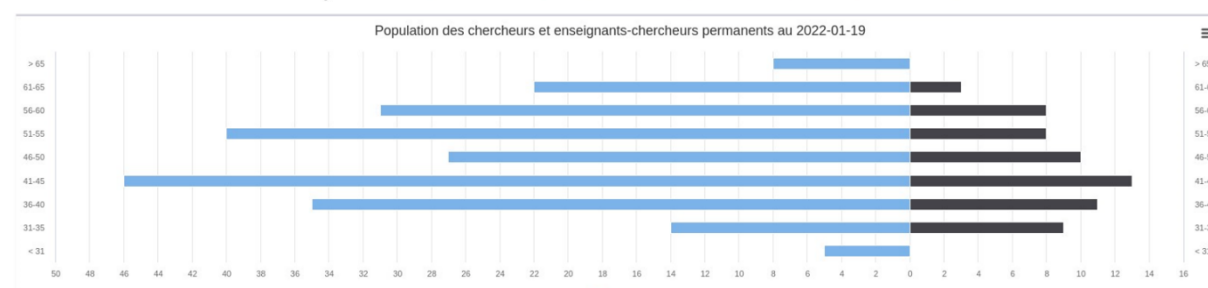
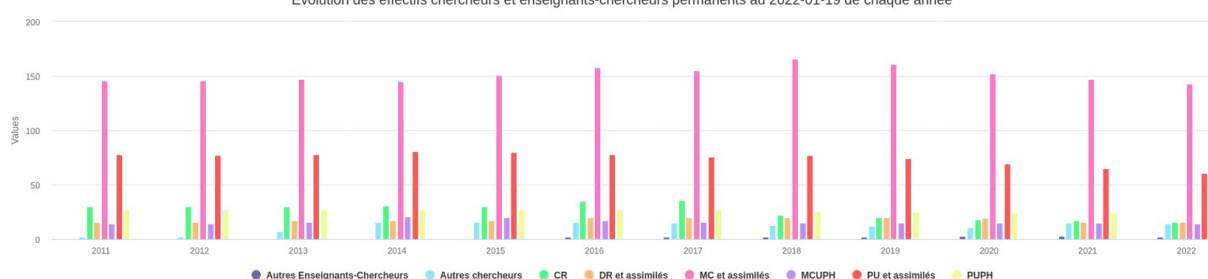
Le rendu de quelques graphiques effectués durant le stage :



Répartition par BAP de l'ensemble des ITA/BIATSS hors associés de l'unité au 2022-01-19



Évolution des effectifs chercheurs et enseignants-chercheurs permanents au 2022-01-19 de chaque année



**Bilan :**

Finalement cette mission m'a permis de développer mes compétences envers le framework symfony. D'apprendre plus sur les possibilités qu'offre les librairies de framework (ici de faire du JavaScript sans avoir besoin de comprendre le langage). J'ai appris à réaliser des tests unitaires.