

SEPTEMBER 17 -18, 2026 - LILLE, FRANCE & ONLINE

API PLATFORM CONFERENCE

**Réinventer les tableaux avec
Symfony, API Platform et
Mercure**

Qui suis-je ?

Tanguy Lemarié

- ✓ **Software Developer / AI Engineer**, Skaleet
- ✓ **Freelance** pour INOXA et ICKO Apiculture



Skaleet

Core Banking solutions built
for speed and control

Plateforme bancaire cloud-native,
API-first et modulaire pour banques,
fintechs et solutions BaaS.

</> API-Centric

Cloud Native

Modular Platform





01

Pourquoi les tableaux sont-ils toujours pénibles ?

02

La promesse d'UX DataTables

03

Pas à pas vers le tableau idéal

04

IA, conclusion et perspective



01

Pourquoi les tableaux sont-ils toujours pénibles ?

02

La promesse d'UX DataTables

03

Pas à pas vers le tableau idéal

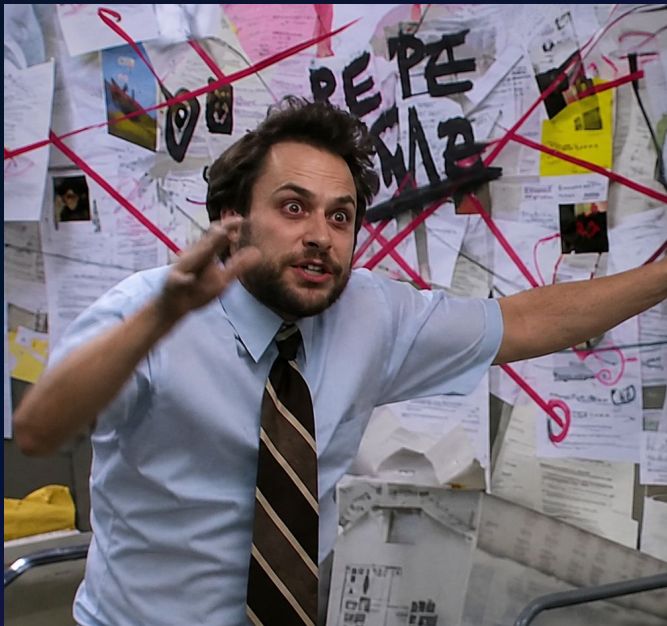
04

IA, conclusion et perspective

Ça commence souvent comme ça



Deux semaines plus tard...



- 👤 Afficher les utilisateurs
- 🔍 Recherche
- ↕ Tri
- ☰ Pagination
- 🔎 Filtres avancés
- 📄 Export CSV / Excel
- ☑ Actions en masse
- 🔒 Permissions
- 📱 Responsive
- ⚡ Temps réel
- 📈 Performance (100k+ utilisateurs)
- ♿ Accessibilité
- 🌐 Internationalisation

Et bien sûr... les changements d'avis ! 😊

Et le pire ? On recommence à chaque projet





01

Pourquoi les tableaux sont-ils toujours pénibles ?

02

La promesse d'UX DataTables

03

Pas à pas vers le tableau idéal

04

IA, conclusion et perspective

Et si on arrêta de réinventer la roue ?

A photograph of a Black man with a mustache, wearing a black leather jacket and a gold watch, resting his chin on his hand in a thoughtful pose. The image is part of a presentation slide with a dark blue background and a faint geometric pattern.

Moins de temps
sur les tableaux,
plus de temps
sur le métier. 🚀

↓

La promesse
d'**UX DataTables**

La promesse d'UX DataTables

UX DataTables

1 bundle

Features:

- Filtres
- Pagination (1 2 3)
- Export CSV / Excel
- SQL
- JavaScript
- API
- Permissions
- Temps réel

Table Data:

ID	Nom	Statut	Date	Montant
#1248	Alice Martin	Actif	12/05/2024	1 250,00 €
#1247	Thomas Durand	Actif	11/05/2024	980,00 €
#1246	Sophie Bernard	En attente	10/05/2024	690,00 €
#1245	Lucas Moreau	Actif	09/05/2024	1 840,00 €
#1244	Chloé Petit	Inactif	08/05/2024	230,00 €

Afficher 10 résultats

Évolution: +23,5%

Répartition: 12 340 € Total

- Actif
- En attente
- Inactif

Logos: Symfony, API Platform, Mercure

Benefits:

- Moins de JavaScript**: Moins de code, plus de valeur.
- Developer Experience**: Productif, fluide, agréable.
- Tout intégré**: Filtres, tri, pagination, export, permissions, temps réel...
- Performance & Scalabilité**: Conçu pour les apps modernes.

La puissance de DataTables, l'expérience Symfony





01

Pourquoi les tableaux sont-ils toujours pénibles ?

02

La promesse d'UX DataTables

03

Pas à pas vers le tableau idéal

04

IA, conclusion et perspective

Votre mission

Le Product Manager vous confie une mission...



Alex, Product Manager 09:42

Bonjour !

Nous avons besoin d'une page de gestion des utilisateurs pour l'équipe d'administration.

Elle doit être puissante, rapide et simple à utiliser.

Merci ! 🙌



Spécifications

- ☐ Afficher la liste des utilisateurs
- ☐ Recherche, tri et pagination
- ☐ Filtrer par rôle et statut
- ☐ Voir les détails et modifier un utilisateur
- ☐ Exporter en CSV
- ☐ Gérer les permissions
- ☐ Supporter de gros volumes (100k+ utilisateurs)
- ☐ Mise à jour en temps réel
- ☐ Responsive et agréable à utiliser



Recherche
& Tri



Filtres



Responsive



Temps réel



Export CSV



Performance



Permissions

Utilisateur	Email	Rôle	Statut	Dernière connexion	Actions
 Sophie Martin	sophie.martin@example.com	ADMIN	● Actif	Il y a 2 min	 
 Thomas Bernard	thomas.bernard@example.com	MANAGER	● Actif	Il y a 15 min	 
 Julie Durand	julie.durand@example.com	USER	● Inactif	Il y a 2 jours	 
 Nicolas Petit	nicolas.petit@example.com	USER	● Actif	Il y a 1 h	 

Les points clés à retenir



UX riche

Recherche, tri, pagination, filtres



Performance

Supporter 100k+ utilisateurs



Permissions

Les bonnes actions
aux bonnes personnes



Temps réel & responsive

Un tableau moderne
jusqu'au bout

Créez votre premier tableau

```
php bin/console make:datatable User
```

UserDataTable.php

```
#[AsDataTable(User::class)]
final class UserDataTable extends AbstractDataTable
{
    public function configureColumns(): iterable
    {
        return [
            NumberColumn::new('id', 'ID'),
            TextColumn::new('email', 'Email'),
            TextColumn::new('firstName', 'First Name'),
            TextColumn::new('lastName', 'Last Name'),
            TextColumn::new('avatar', 'Avatar'),
            BooleanColumn::new('active', 'Active'),
            DataColumn::new('lastLoginAt', 'Last Login At'),
            DataColumn::new('createdAt', 'Created At'),
            DataColumn::new('updatedAt', 'Updated At'),
        ];
    }
}
```

Afficher le tableau



UserController.php

```
#[Route('/users', name: 'app_users_list')]
public function index(UserDataTable $table): Response
{
    return $this->render('users/list.html.twig', [
        'table' => $table
    ]);
}
```



list.html.twig

```
{% block body %}
    <div class="min-h-screen bg-gray-50 px-4 py-10 dark:bg-gray-900">
        <div class="mx-auto w-full max-w-6xl">
            {{ render_datatable(table, {
                'class': '...'
            }) }}
        </div>
    </div>
{% endblock %}
```

Notre premier résultat

Afficher 10 entrées Rechercher...

	ID	Email	First Name	Last Name	Avatar	Active	Last Login At	Created At	Updated At
	210494	user-000001@example.com	Bob	Smith	https://i.pravatar.cc/128?u=user-000001	true	2024-09-10	2024-09-10	2024-09-11
	210495	user-000002@example.com	Charlie	Smith	https://i.pravatar.cc/128?u=user-000002	true	2024-09-10	2024-09-10	2024-09-11
	210496	user-000003@example.com	Diana	Smith	https://i.pravatar.cc/128?u=user-000003	true	2024-09-10	2024-09-10	2024-09-11
	210497	user-000004@example.com	Emma	Smith	https://i.pravatar.cc/128?u=user-000004	true	2024-09-10	2024-09-10	2024-09-11
	210498	user-000005@example.com	Frank	Smith	https://i.pravatar.cc/128?u=user-000005	true		2024-09-10	2024-09-11
	210499	user-000006@example.com	Grace	Smith	https://i.pravatar.cc/128?u=user-000006	true	2024-09-10	2024-09-10	2024-09-11
	210500	user-000007@example.com	Henry	Smith		true	2024-09-10	2024-09-10	
	210501	user-000008@example.com	Iris	Johnson		true	2024-09-10	2024-09-10	
	210502	user-000009@example.com	Jack	Johnson		true	2024-09-10	2024-09-10	
	210503	user-000010@example.com	Kate	Johnson	https://i.pravatar.cc/128?u=user-000010	true		2024-09-10	

Affichage de 1 à 10 sur 100 000 entrées

« < 1 2 3 4 5 ... 10 000 > »

Personnaliser les colonnes

```
UserDataTable.php

public function configureColumns(): iterable
{
    yield TemplateColumn::new('avatar', 'Utilisateur')
        →setTemplate('user/data_table/user.html.twig');

    yield EmailColumn::new('email', 'Email')
        →setClassName('whitespace-nowrap')
        →renderAsText();

    yield ChoiceColumn::new('role', 'Rôle')
        →setChoices(UserRole::class)
        →renderAsBadges(UserRole::badgeVariants());

    yield BooleanColumn::new('active', 'Statut')
        →renderAsSwitch();

    yield DateColumn::new('lastLoginAt', 'Dernière connexion')
        →setDefaultContent('Jamais')
        →relative();
}
```

Colonne Utilisateur

```
user.html.twig

{% set first_name = row.firstName|default('') %}
{% set last_name = row.lastName|default('') %}
{% set full_name = row.fullName %}

<div class="user-table-person">
    {% if row.avatar is not null %}
        
    {% else %}
        <span class="user-table-avatar user-table-avatar-fallback">
            {{ first_name|slice(0, 1)|upper }}{{ last_name|slice(0, 1)|upper }}
        </span>
    {% endif %}

    <span class="user-table-name">
        {{ full_name|default(row.email) }}
    </span>
</div>
```



Bob Smith



Henry Smith

Le tableau prend forme

Afficher 10 entrées

Utilisateur	Email	Rôle	Statut	Dernière connexion
 Bob Smith	user-000001@example.com	Admin	☒	Il y a 730 jours
 Charlie Smith	user-000002@example.com	Support	☒	Il y a 730 jours
 Diana Smith	user-000003@example.com	User	☒	Il y a 730 jours
 Emma Smith	user-000004@example.com	Manager	☒	Il y a 730 jours
 Frank Smith	user-000005@example.com	Support	☒	A l'instant
 Grace Smith	user-000006@example.com	User	☒	Il y a 730 jours
HS  Henry Smith	user-000007@example.com	Manager	☒	Il y a 730 jours
IJ  Iris Johnson	user-000008@example.com	Support	☒	Il y a 730 jours
JJ  Jack Johnson	user-000009@example.com	User	☒	Il y a 730 jours
 Kate Johnson	user-000010@example.com	Manager	☒	A l'instant

Affichage de 1 à 10 sur 100 000 entrées

« < 1 2 3 4 5 ... 10 000 > »

Et avec 100 000 utilisateurs ?

Query Metrics

Database Queries	Different statements	Query time	Invalid entities	Managed entities
3	2	169.08 ms	0	100000

Queries Database Connections Entity Managers Second Level Cache Managed Entities Entities Mapping

Group similar statements

#▲	Time	Info
1	56.92 ms	SELECT COUNT(DISTINCT u0_.id) AS sclr_0 FROM "user" u0_ Parameters: [] View formatted query View runnable query Explain query View query backtrace
2	12.53 ms	SELECT COUNT(DISTINCT u0_.id) AS sclr_0 FROM "user" u0_ Parameters: [] View formatted query View runnable query Explain query View query backtrace
3	99.63 ms	SELECT u0_.id AS id_0, u0_.email AS email_1, u0_.status AS status_2, u0_.first_name AS first_name_3, u0_.last_name AS last_name_4, u0_.avatar AS avatar_5, u0_.role AS role_6, u0_.active AS active_7, u0_.last_login_at AS last_login_at_8, u0_.created_at AS created_at_9, u0_.updated_at AS updated_at_10 FROM "user" u0_ Parameters: [] View formatted query View runnable query Explain query View query backtrace

Performance metrics

Total execution time	Symfony initialization	Peak memory usage
2401ms	14ms	463.80MiB

Passer en mode server-side



php

UserDataTable.php

```
public function configureDataTable(DataTable $table): DataTable
{
    return $table
        →serverSide();
}
```

Le même tableau, mais

Afficher 10 entrées

Rechercher...

Utilisateur	Email	Rôle	Statut	Dernière connexion
 Bob Smith	user-000001@example.com	Admin	☒	Il y a 730 jours
 Charlie Smith	user-000002@example.com	Support	☒	Il y a 730 jours
 Diana Smith	user-000003@example.com	User	☒	Il y a 730 jours
 Emma Smith	user-000004@example.com	Manager	☒	Il y a 730 jours
 Frank Smith	user-000005@example.com	Support	☒	A l'instant
 Grace Smith	user-000006@example.com	User	☒	Il y a 730 jours
HS Henry Smith	user-000007@example.com	Manager	☒	Il y a 730 jours
IJ Iris Johnson	user-000008@example.com	Support	☒	Il y a 730 jours
JJ Jack Johnson	user-000009@example.com	User	☒	Il y a 730 jours
 Kate Johnson	user-000010@example.com	Manager	☒	A l'instant

Affichage de 1 à 10 sur 100 000 entrées

« < 1 2 3 4 5 ... 10 000 > »

100 000 lignes, 6 Mo de

Query Metrics

Database Queries	Different statements	Query time	Invalid entities	Managed entities
3	2	46.41 ms	0	10

Queries Database Connections Entity Managers Second Level Cache Managed Entities Entities Mapping

Group similar statements

#▲	Time	Info
1	36.14 ms	SELECT COUNT(DISTINCT u0_.id) AS sclr_0 FROM "user" u0_ Parameters: [] View formatted query View runnable query Explain query View query backtrace
2	7.32 ms	SELECT COUNT(DISTINCT u0_.id) AS sclr_0 FROM "user" u0_ Parameters: [] View formatted query View runnable query Explain query View query backtrace
3	2.95 ms	SELECT u0_.id AS id_0, u0_.email AS email_1, u0_.status AS status_2, u0_.first_name AS first_name_3, u0_.last_name AS last_name_4, u0_.avatar AS avatar_5, u0_.role AS role_6, u0_.active AS active_7, u0_.last_login_at AS last_login_at_8, u0_.created_at AS created_at_9, u0_.updated_at AS updated_at_10 FROM "user" u0_ LIMIT 10 Parameters: [] View formatted query View runnable query Explain query View query backtrace

Performance metrics

Total execution time	Symfony initialization	Peak memory usage
87ms	23ms	6.00MiB

Connecter le tableau à API Platform

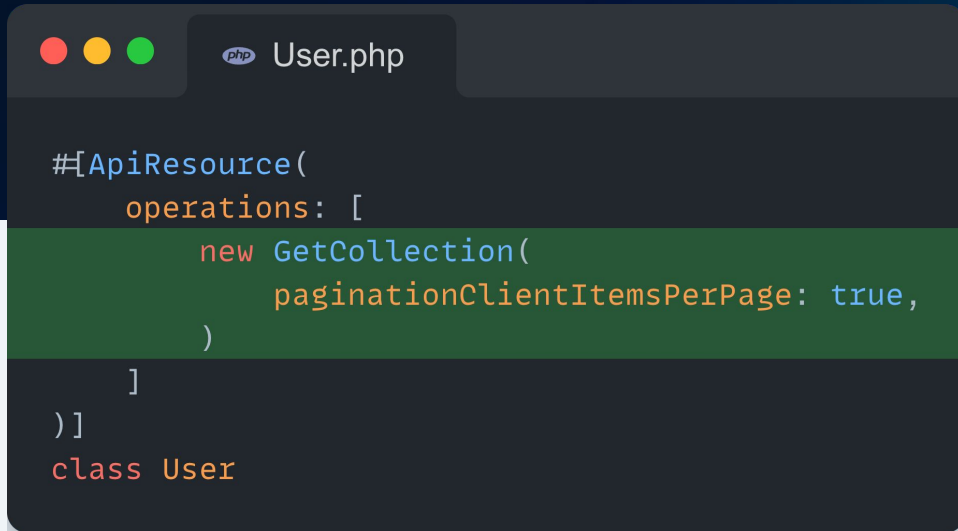


php

UserDataTable.php

```
#[AsDataTable(  
    entityClass: User::class,  
    apiPlatform: true  
)]  
final class UserDataTable extends AbstractDataTable
```

Exposer les utilisateurs avec API Platform

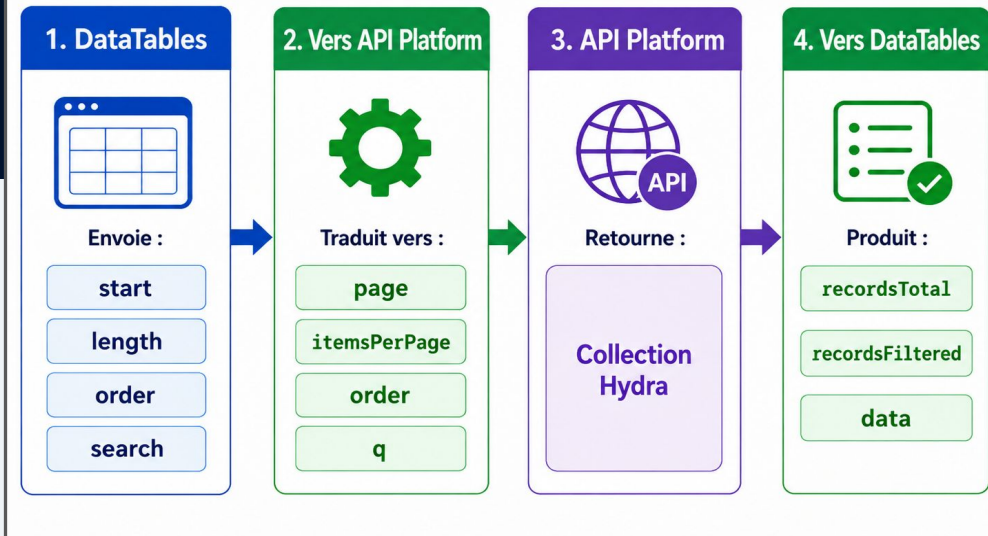


```
#[ApiResponse(  
    operations: [  
        new GetCollection(  
            paginationClientItemsPerPage: true,  
        )  
    ]  
)]  
class User
```

L'adapter fait le lien

Api Platform Adapter

Pont entre DataTables (mode server-side) et API Platform



Langue, tri et pagination

```
UserDataTable.php

use Pentiminax\UX\DataTables\Enum\Language;

public function configureDataTable(DataTable $table): DataTable
{
    return $table
        →language(Language::FR)
        →lengthMenu([10, 25, 50, 100])
        →order([[0, 'desc']])
        →pageLength(25);
}
```

Affiche entrées

10
✓ 25
50

Affichage de 1 à 25 sur 100 000 entrées

Configurer les filtres

UserDataTable.php

```
use Pentiminax\UX\DataTables\Filter\ChoiceFilter;
use Pentiminax\UX\DataTables\Filter\CheckboxFilter;
use Pentiminax\UX\DataTables\Filter\DateRangeFilter;
use Pentiminax\UX\DataTables\Model\Filters;

public function configureFilters(Filters $filters): Filters
{
    $roleFilter = ChoiceFilter::new('role')
        →label('Rôle')
        →options(UserRole::cases());

    $statusFilter = CheckboxFilter::new('active')
        →label('Actif');

    $lastLoginAtFilter = DateRangeFilter::new('lastLoginAt')
        →label('Dernière connexion');

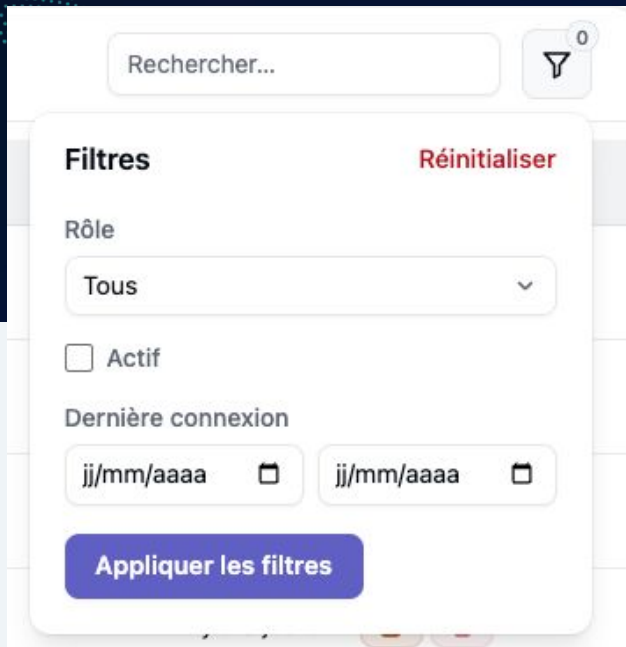
    return $filters
        →add($roleFilter)
        →add($statusFilter)
        →add($lastLoginAtFilter);
}
```

Exposer les filtres avec API

```
User.php

#[ApiResponse(
    operations: [
        new GetCollection(
            parameters: [
                'role' => new QueryParameter(
                    filter: new BackedEnumFilter(),
                    property: 'role'
                ),
                'active' => new QueryParameter(
                    filter: new BooleanFilter(),
                    property: 'active'
                ),
                'lastLoginAt' => new QueryParameter(
                    filter: new DateFilter(),
                    property: 'lastLoginAt'
                ),
            ],
        )
    ]
)]
class User
```

Des filtres prêts à l'emploi



The image shows a user interface for a search and filter system. At the top, there is a search bar with the placeholder text "Rechercher...". To its right is a filter icon (a funnel) with a small circle containing the number "0". Below the search bar, a modal window titled "Filtres" is displayed. The modal has a "Réinitialiser" (Reset) link in red text. Inside the modal, there are three filter sections: 1. "Rôle" (Role) with a dropdown menu currently set to "Tous" (All). 2. "Actif" (Active) with an unchecked checkbox. 3. "Dernière connexion" (Last connection) with two date input fields, each containing the placeholder "jj/mm/aaaa" and a calendar icon. At the bottom of the modal is a large blue button labeled "Appliquer les filtres" (Apply filters).

Rechercher...

Filtres [Réinitialiser](#)

Rôle

Tous

☐ Actif

Dernière connexion

jj/mm/aaaa  jj/mm/aaaa 

Appliquer les filtres

Exporter les données du

```
UserDataTable.php

use Pentiminox\UX\DataTables\Model\Extensions\Button;

public function configureDataTable(DataTable $table): DataTable
{
    $csvButton = Button::csv(
        serverSide: true
    );

    $excelButton = Button::excel(
        serverSide: true
    );

    return $table
        →buttons([$csvButton, $excelButton]);
}
```

Exporter les données du tableau

Afficher	25	entrées	CSV	Excel	Rechercher...	0
Utilisateur	Email	Rôle	Statut	Dernière connexion	Actions	

Voir les détails d'un utilisateur

```
UserDataTable.php

use Pentiminax\UX\DataTables\Model\Action;
use Pentiminax\UX\DataTables\Model\Actions;

public function configureActions(Actions $actions): Actions
{
    $detailAction = Action::detail(label: '')
        ->icon('fa-solid fa-eye')
        ->linkToRoute(
            'app_user_show',
            fn(User $user): array => ['id' => $user->getId()]
        )
        ->htmlAttributes(['target' => '_blank']);

    return $actions->add($detailAction);
}
```

Utilisateur	Email	Rôle	Statut	Dernière connexion	Actions
 Bob Smith	user-000001@example.com	Admin		Il y a 730 jours	 

Modifier un utilisateur



```
UserDataTable.php

use Pentiminax\UX\DataTables\Model\Action;
use Pentiminax\UX\DataTables\Model\Actions;

public function configureActions(Actions $actions): Actions
{
    $editAction = Action::edit(label: '')
        →icon('fa-solid fa-pen-to-square');

    return $actions→add($editAction);
}
```

Edit Bob Smith

Email

user-000001@example.com

Rôle

Admin

Statut

☐

Dernière connexion

11/09/2026

Cancel

Save

Restreindre une action

```
UserDataTable.php

use Pentiminox\UX\DataTables\Model\Action;
use Pentiminox\UX\DataTables\Model\Actions;

public function configureActions(Actions $actions): Actions
{
    $deleteAction = Action::delete(label: '')
        →disabledWhenDenied()
        →icon('fa-solid fa-trash')
        →setPermission('ROLE_ADMIN');

    return $actions→add($deleteAction);
}
```



Centraliser les permissions avec un

```
UserVoter.php

use Pentiminox\UX\DataTables\Security\Permission;

class UserVoter extends Voter
{
    public function __construct(
        private readonly AccessDecisionManagerInterface $accessDecisionManager,
    ) {
    }

    protected function supports(string $attribute, mixed $subject): bool
    {
        return \in_array($attribute, [
            Permission::DT_EDIT_ROW,
            Permission::DT_DELETE_ROW,
            Permission::DT_VIEW_ROW_DETAILS,
        ], true) && $subject instanceof User;
    }

    protected function voteOnAttribute(
        string $attribute,
        mixed $subject,
        TokenInterface $token,
        ?Vote $vote = null
    ): bool {
        return match($attribute) {
            Permission::DT_EDIT_ROW,
            Permission::DT_VIEW_ROW_DETAILS => true,
            Permission::DT_DELETE_ROW =>
                $this->accessDecisionManager->decide(
                    $token,
                    ['ROLE_ADMIN']
                ),
            default => false,
        };
    }
}
```

Activer le temps réel avec Mercure



php

UserDataTable.php

```
#[AsDataTable(  
    entityClass: User::class,  
    mercure: true,  
    apiPlatform: true,  
)]  
final class UserDataTable extends AbstractDataTable
```

Mettre en évidence les mises à jour



php

UserDataTable.php

```
public function configureDataTable(DataTable $table): DataTable
{
    return $table
        →highlightUpdates(
            ignoreColumns: ['lastLoginAt']
        )
}
```

Le temps réel en action

Utilisateurs

Recherchez, filtrez et exportez les comptes de la plateforme.

Afficher

25

entrées

CSV

Excel

Rechercher...



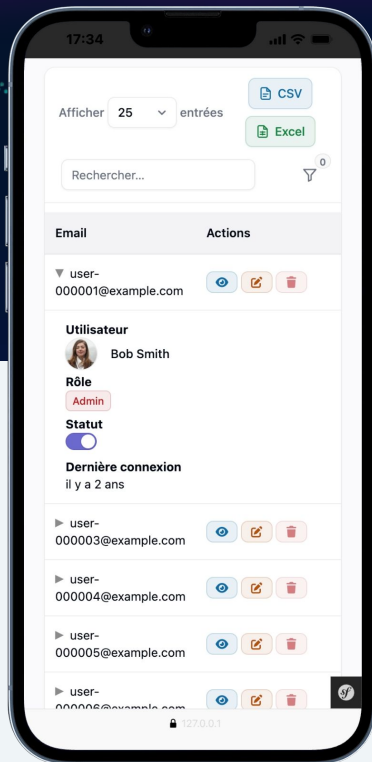
Utilisateur	Email	Rôle	Statut	Dernière connexion	Actions
 Bob Smith	user-000001@example.com	Admin	☐	2 minutes ago	  
 Diana Smith	user-000003@example.com	Utilisateur	☒	3 minutes ago	  
 Emma Smith	user-000004@example.com	Manager	☒	2 minutes ago	  
 Frank Smith	user-000005@example.com	Support	☒	2 minutes ago	  
 Grace Smith	user-000006@example.com	Utilisateur	☒	2 minutes ago	  
HS  Henry Smith	user-000007@example.com	Manager	☒	3 minutes ago	  
IJ  Iris Johnson	user-000008@example.com	Support	☐	2 minutes ago	  
 Jack Johnson	user-		☐		  

Activer le responsive

  UserDataTable.php

```
public function configureDataTable(DataTable $table): DataTable
{
    return $table
        →responsive();
}
```

Le tableau s'adapte au mobile



Le tableau final

Afficher 10 entrées CSV Excel

Rechercher...

Utilisateur	Email	Rôle	Statut	Dernière connexion	Actions
 Bob Smith	user-000001@example.com	Admin	☒	il y a 2 ans	  
 Diana Smith	user-000003@example.com	Utilisateur	☒	il y a 2 ans	  
 Emma Smith	user-000004@example.com	Manager	☒	il y a 2 ans	  
 Frank Smith	user-000005@example.com	Support	☒	Jamais	  
 Grace Smith	user-000006@example.com	Utilisateur	☒	il y a 2 ans	  
HS  Henry Smith	user-000007@example.com	Manager	☒	il y a 2 ans	  
IJ  Iris Johnson	user-000008@example.com	Support	☒	il y a 2 ans	  
JJ  Jack Johnson	user-000009@example.com	Utilisateur	☒	il y a 2 ans	  
 Kate Johnson	user-000010@example.com	Manager	☒	Jamais	  
 Leo Johnson	user-000011@example.com	Support	☒	il y a 2 ans	  

Affichage de 1 à 10 sur 99 999 entrées

« < 1 2 3 4 5 ... 10 000 > »



01

Pourquoi les tableaux sont-ils toujours pénibles ?

02

La promesse d'UX DataTables

03

Pas à pas vers le tableau idéal

04

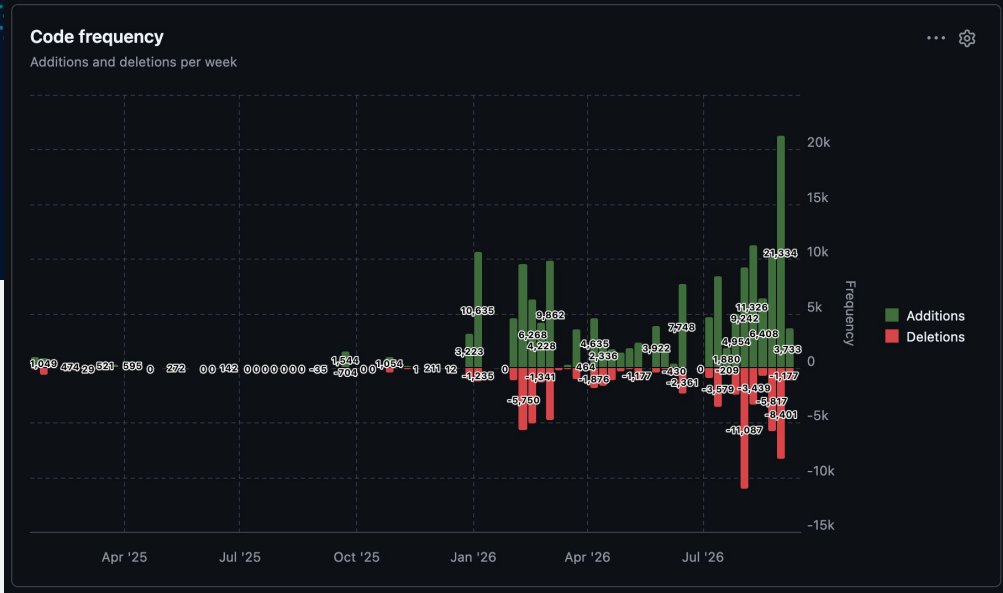
IA et Conclusion

Je ne développe plus ce bundle "seul"



Et ça se voit dans le rythme du projet

Code frequency over the history of pentiminax/ux-datatables



UX DataTables v1.0

DATA WORKS BETTER TOGETHER

Same Tables
Bigger Possibilities



Filtering



Sorting



Responsive

Search users, email, role...

All statuses

All roles

Last 30 days

+ New

#	Name	Email	Role	Status	Last activity	Actions
101	Emma Laurent	emma@acme.dev	Admin	Online	Just now Updated	
102	Lucas Moreau	lucas@acme.dev	Developer	Online	2 min ago	
103	Sophie Martin	sophie@acme.dev	Product	Away	12 min ago	
104	Noah Dubois	noah@acme.dev	Designer	Offline	1 hour ago	
105	Chloé Bernard	chloe@acme.dev	Developer	Online	Just now Updated	
106	Thomas Petit	thomas@acme.dev	Analyst	Online	5 min ago	

Showing 1 to 6 of 248 entries

<< 1 2 3 4 5 ... 42 >>

10 per page



Mercure v1.0
support

real-time updates

- Emma Laurent updated Just now
- Chloé Bernard updated Just now
- New user registered 1 min ago
- Status changed 2 min ago



Export



API Platform

CLEAN DATA
HAPPIER PEOPLE
BRIGHTER TOMORROW

BUILD
ANALYZE
REAL-TIME
CELEBRATE

BETTER DATA
BRIGHTER APPS
TOGETHER

v1.0
AT LAST!

Open
for a brighter
web

Merci !

Des questions ?

FOLLOW ME!

X @pentiminax

github.com/pentiminax/ux-datatables

