

# Unpoly

# pour reprendre le contrôle !

ESTÉBAN SOUBIRAN

Devoxx · web

12 avril 2023


# Estéban Soubiran





Élève ingénieur en cybersécurité

Développeur web chez Aneo

Responsable du Classement des Associations

 [esteban-soubiran.site](https://esteban-soubiran.site)

 [Barbapapazes](#)

 [Estéban Soubiran](#)

# Notre fil rouge



# La plateforme de vote : spécifications



- Visualiser, ajouter et modifier les associations qui participent au Classement (textes et fichiers)
- Permettre à chacun de voter avec une adresse mail
- Visualiser sur des graphiques le nombre de vote par jour
- Être SEO friendly
- Avoir une expérience utilisateur rapide et moderne
- Être maintenable et peu complexe

 100 000 visites

 25 000 votes

 45 000 mails

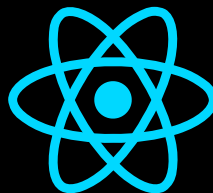
# La stack d'Atom : MERN



MongoDB



Express



React



Node.js

# Le parcours d'Atom - API JSON



## Dépendances

Trouver et gérer les bons paquets



## Modèles

Définir la structure des données



## Routage

Définir les routes de l'application



## API

Gérer les requêtes (filtres, pagination, etc.)



## Authentification

Sécuriser les accès à l'API



## Autorisations

Gérer qui peut accéder à quoi



## Production

Rendre l'application disponible

# Le parcours d'Atom - **Front-end modern**



## **Dépendances**

Trouver et gérer les bons paquets



## **Routage**

Définir les routes de l'application



## **API**

Récupération des données à l'API



## **Autorisations**

Gérer qui peut accéder à quoi



## **Accessibilité**

Tout le monde doit pouvoir l'utiliser



## **SEO**

Optimiser le référencement



## **Production**

Rendre l'application disponible

# Les 3 problèmes d'Atom



Temps limité



Complexité trop élevée



Maintenabilité difficile



# Unpoly

The unobtrusive JavaScript framework  
for server-side web applications

# Ce que les applications rendus côté serveur **font de bien**

- Large choix d'outils matures et éprouvés
- Technologie peu complexe
- Accès aux données de manière synchrone
- Temps du premier rendu (FCP)
- Fonctionne sur tous les types d'appareils

# Ce que les applications rendus côté serveur **ne font pas bien**

1. Retour d'information sur l'interaction lente
2. Chargement des pages détruit les états transitoires
3. Interactivité limitée notamment sur la gestion des "layers"
4. Animations complexes
5. Gestion des formulaires complexes

# Unpoly **change la donne**



25  
**attributs HTML**



Logique  
**serveur centrée**



Supporte  
**tous les langages**



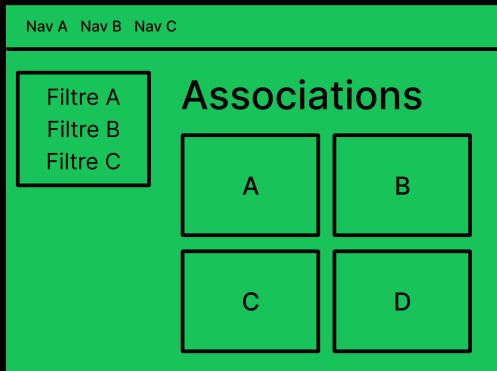
Gère la  
**navigation,**  
les **formulaires,**  
les **animations**  
et les **interactions**

# Démo

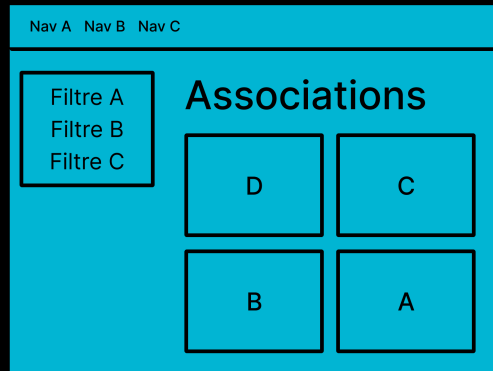
1. Installer Unpoly
2. Découverte des attributs `up-target`, `up-instant`, `up-preload`
3. Gestion des formulaires avec l'attribut `up-submit`
4. Création d'une modal avec l'attribut `up-layer`
5. Personnalisation de la modal
6. Mise en situation de `up.compiler`
7. Inspection des requêtes avec le `server protocol`

# Flux classique des pages

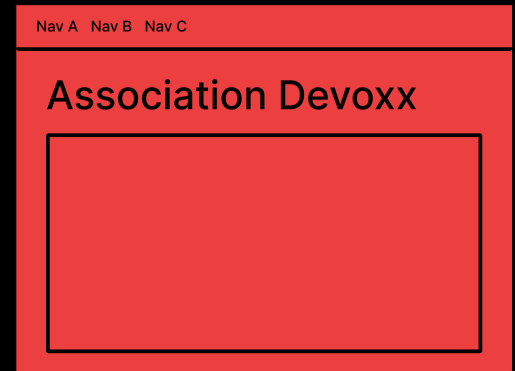
/associations



/associations?order=-1

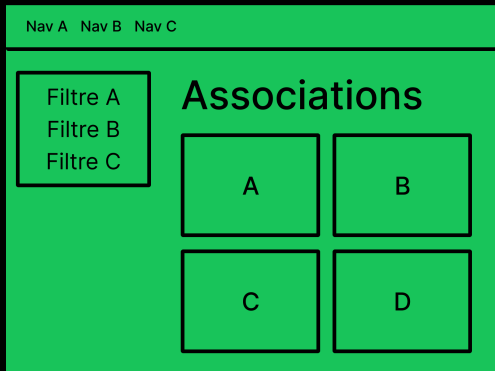


/associations/devoxx

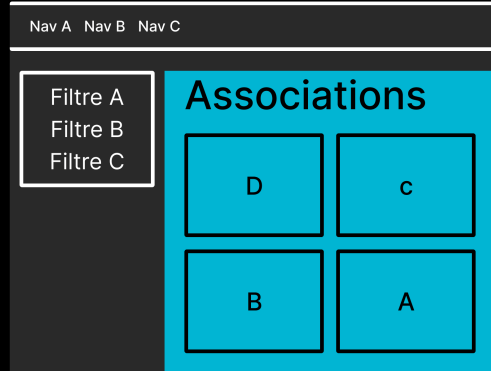


# Flux Unpoly des pages

Chargement initial



up-target="[grid]"



up-target="[layout-main]"



# Amélioration progressive

Progressive enhancement





Le **web**, ce n'est pas si compliqué !

(avec les bons outils)

Merci

# Sources

- [Site officiel d'Unpoly](#)
- [We're breaking up with JavaScript Frontends](#) par Henning Koch
- [Have Single-Page Apps Ruined the Web?](#) par Rich Harris
- [Everyone has JavaScript](#) par Stuart Langridge
- [Progressive enhancement](#) par Wikipedia
- [Polls App](#) par AdonisJS
- [Remix Jokes](#) par AdonisJS
- [Harminder Virk](#)
- [Romain Lanz](#)