

Programmation Web – Avancé

JavaScript & Node.js

Partie 20 : SPA monolithique & routage des composants

Version 2020



Attribution –
Partage dans les
Mêmes Conditions
4.0 International
(CC BY-SA 4.0)

*Presentation template
by [SlidesCarnival](#)*



Création d'une Single Page Application (SPA) monolithique

Et si l'on chargeait un contenu de façon asynchrone...



Qu'est-ce qu'une SPA ?

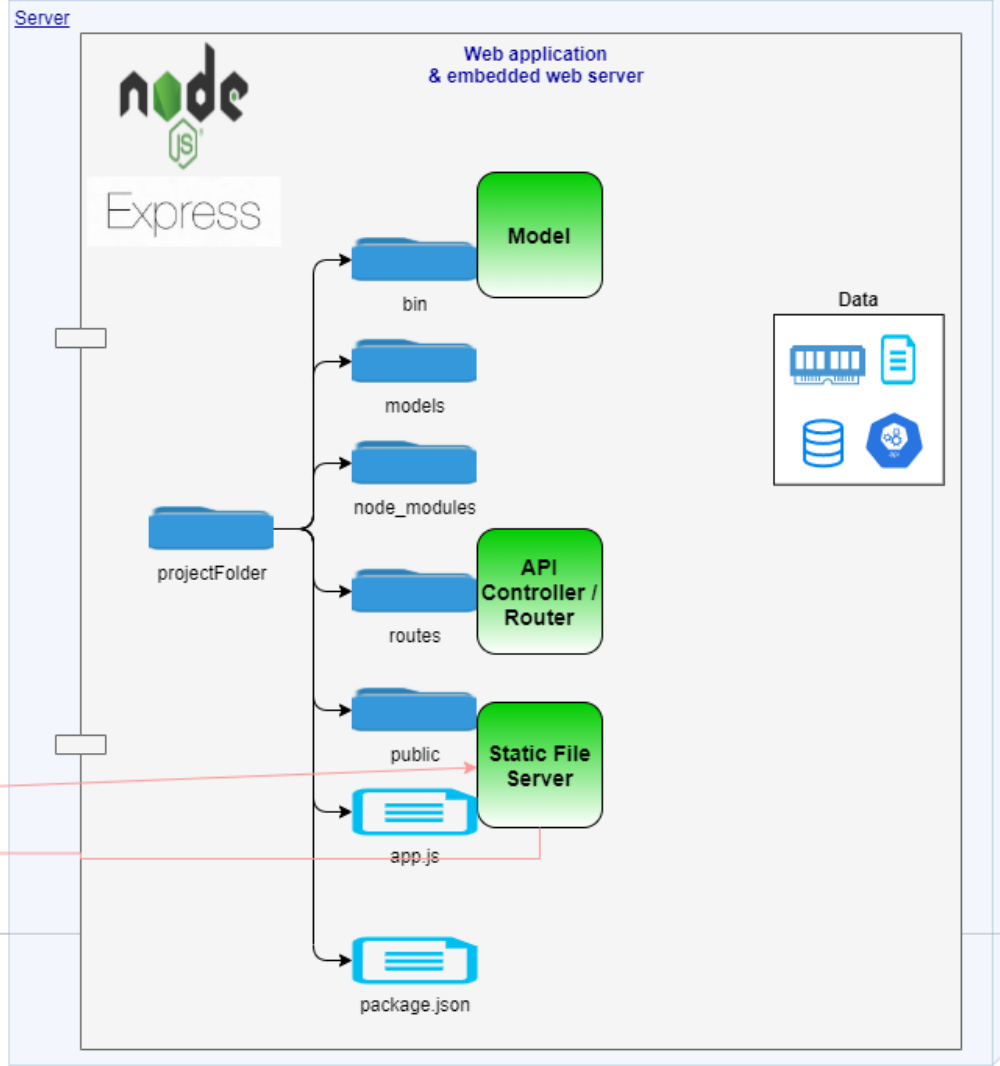
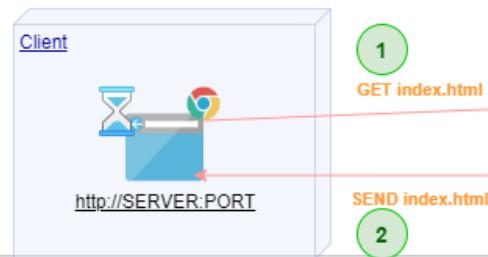
- Pas de rechargement de page
- Réécriture dynamique
- Expérience utilisateur augmentée



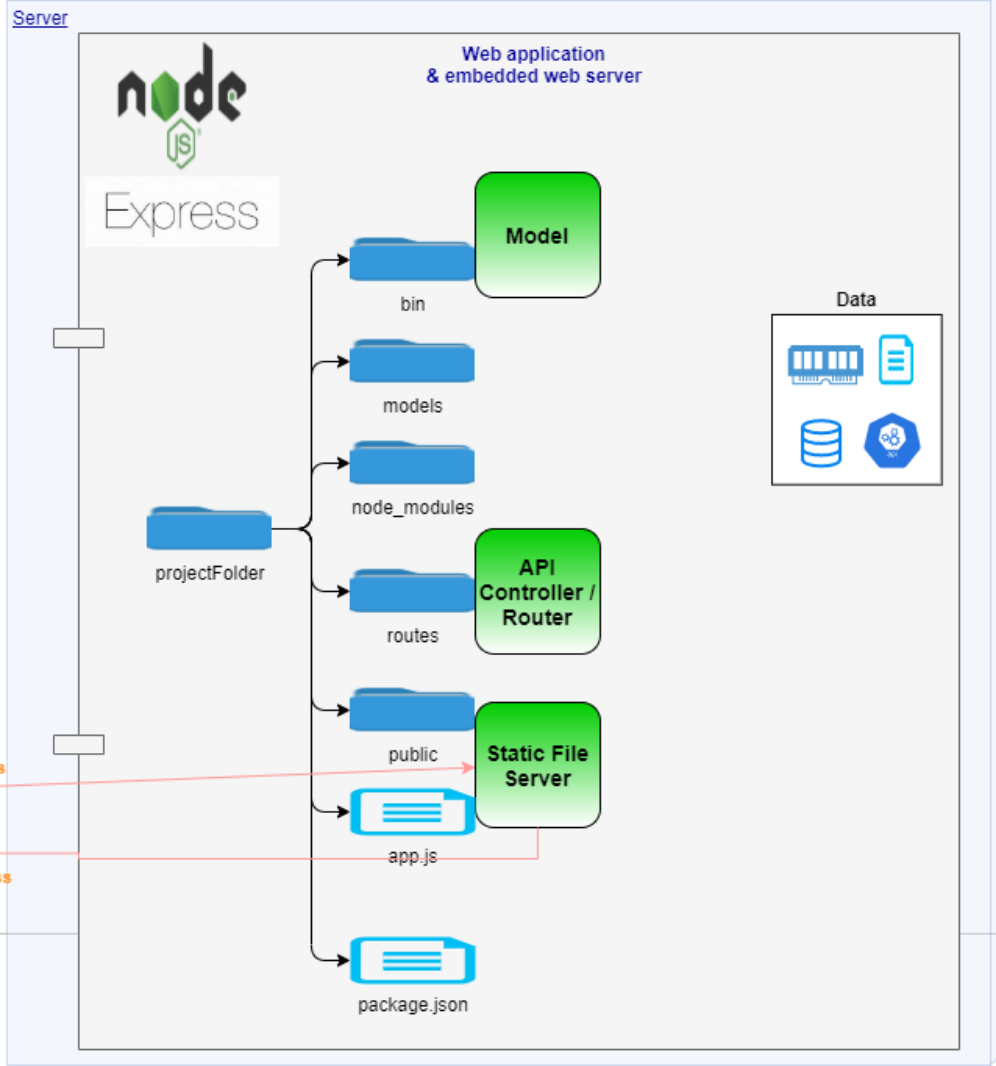
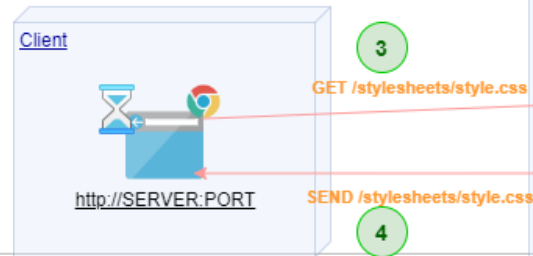
Fonctionnement général d'une SPA

- A) Frontend & Backend : Chargement et rendu du frontend

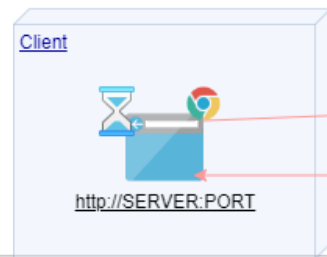
SPA monolithique sous Express : chargement du frontend



SPA monolithique sous Express : chargement du frontend



SPA monolithique sous Express : chargement du frontend



Server



Web application
& embedded web server

Express

projectFolder

bin

models

node_modules

routes

public

app.js

package.json

Model

API
Controller /
Router

Static File
Server

Data



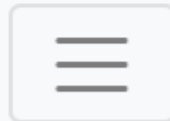


SPA monolithique sous Express : rendu du frontend

JavaScript & Node.js full course



MyCMS



Home

Register

Login

Demo : Monolith SPA

Welcome to MyCMS...



Fonctionnement général d'une SPA

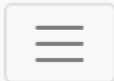
- B) Frontend : Gestion de l'affichage des composants de la page suite aux actions des utilisateurs



SPA monolithique sous Express : demande d'affichage du composant de Login

JavaScript & Node.js full course

MyCMS



Home

Register

Login



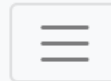
Demo : Monolith SPA

Welcome to MyCMS...

Comment implémenter le
nouvel affichage ?

JavaScript & Node.js full course

MyCMS



Home

Register

Login

Demo : Monolith SPA

Email

johndoe

Password

.....

Submit



SPA : gestion de l'affichage des composants de la page

- Pas de standards
- Afficher / cacher :
 - des éléments d'un menu
 - des éléments d'une page
 - ...
- Problème :
 - pas d'historique !
 - pas d'URL spécifique quand clic sur un menu item



Possibilités pour la gestion de l'affichage de composants de la page

- Simple gestion de containers & HTML dynamique (via template literals)
- Web Components API [\[69.\]](#) :
 - Création d'éléments personnalisés réutilisables
 - Création de HTML templates
 - ...



Possibilités pour la gestion de l'affichage de composants de la page

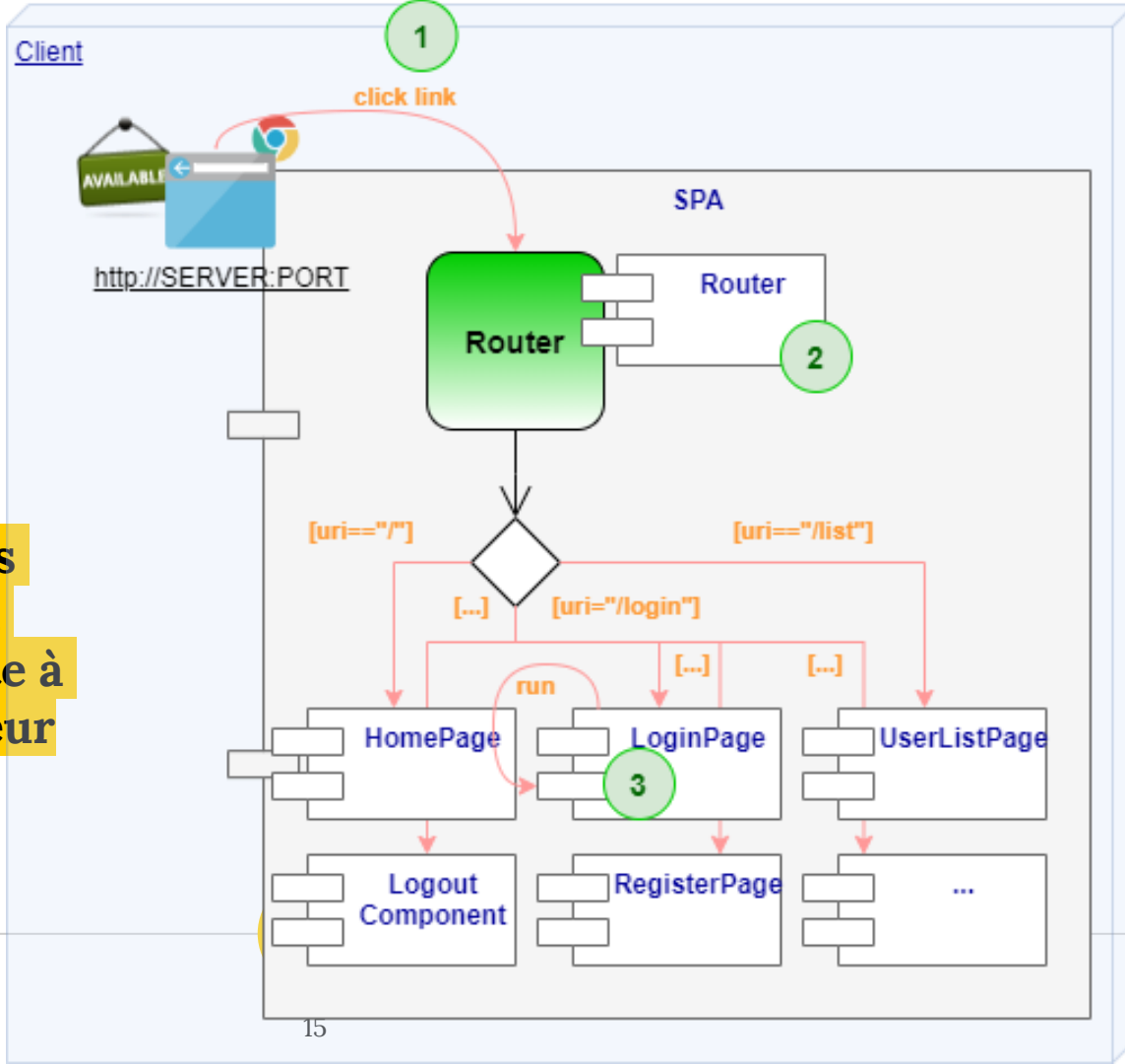
- Utilisation d'un langage de templating : Handlebars [\[67.\]](#), Mustache...
- Utilisation de bibliothèques légères : lit-html [\[70.\]](#), underscore [\[71.\]](#)
- Utilisation de frameworks / bibliothèques : React, Vue...



Conseils pour une simple gestion de l'affichage

- Structure de votre SPA en composants :
navBar, listPage, loginPage, registerPage...
- Routage : afficher / cacher les “composants” lors d’un clic
- Structurer votre JS côté client : création et utilisation des modules JS :
 - portant le nom de vos composants (listPage.js, loginPage.js...)
 - pour des fonctions de type “utils”

SPA monolithique sous Express : affichage du composant de Login suite à une action d'un utilisateur





SPA monolithique sous Express : demande d'affichage de la liste d'utilisateurs

JavaScript & Node.js full
course

MyCMS



Home

List

Logout

gabor@it.hu



Début d'exécution du composant :
demande des données à une web
API

JavaScript & Node.js
full course

MyCMS



Demo : Monolith SPA

List of MyCMS users

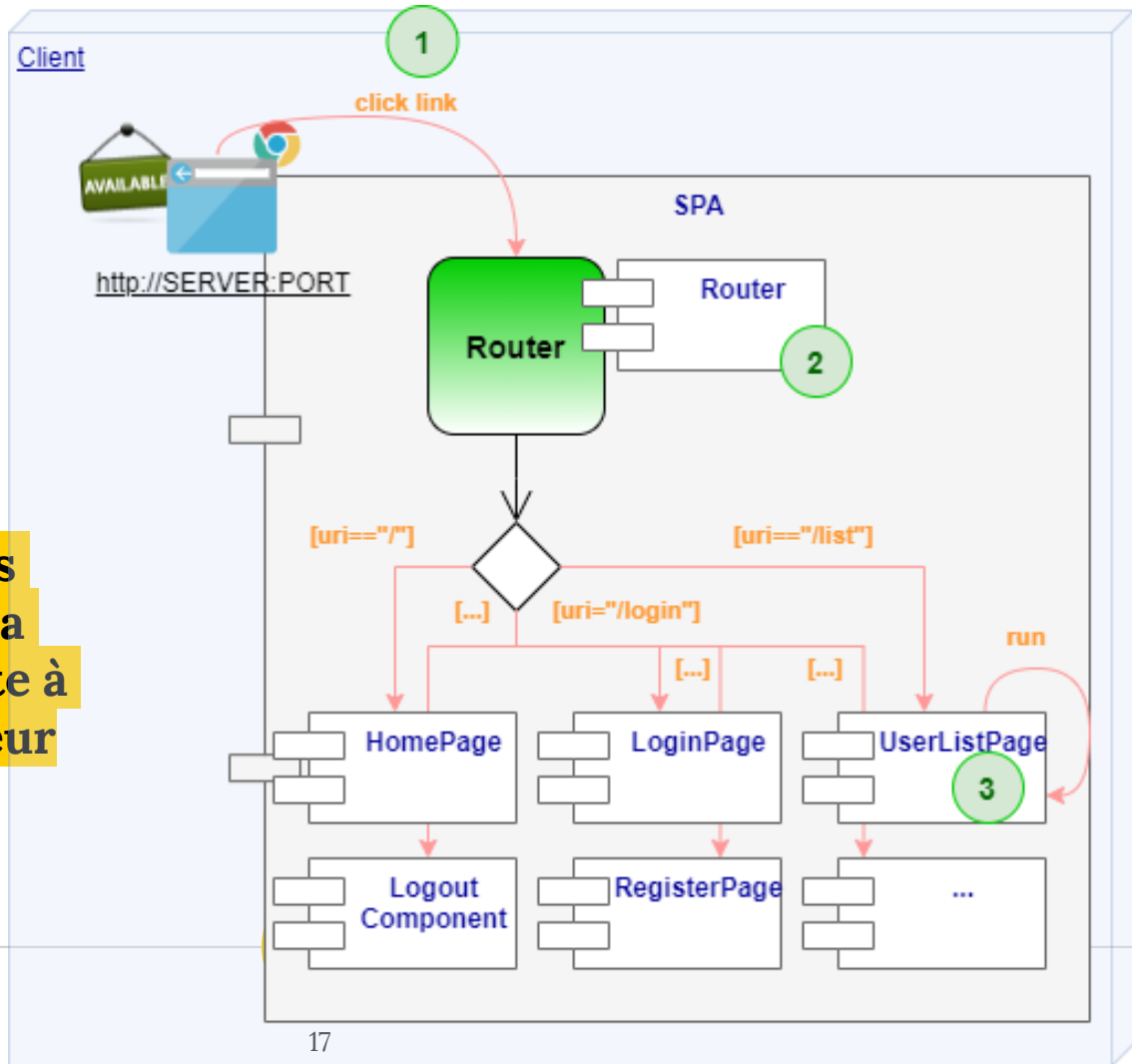
raoulette@gmail.com

teacher@vinci.be

voila@ici.be

gabor@it.hu

SPA monolithique sous Express : affichage de la liste des utilisateurs suite à une action d'un utilisateur

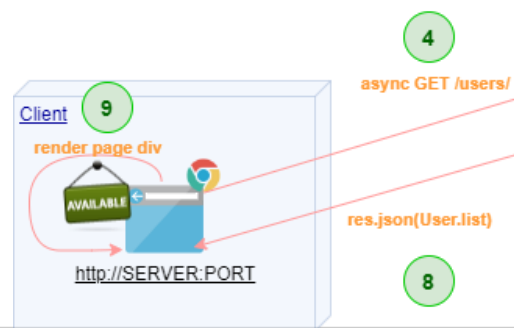




Fonctionnement général d'une SPA

- C) Frontend : Appel asynchrone d'une opération d'une Web API
- D) Web API : Gestion de l'opération sur les ressources et réponse au frontend (voir Partie 19 concernant les RESTful API)
- E) Frontend : Gestion de la réponse et éventuelle MàJ de l'affichage de la SPA

SPA monolithique sous Express : gestion de l'appel à une web API pour obtenir la liste des utilisateurs



4

async GET /users/

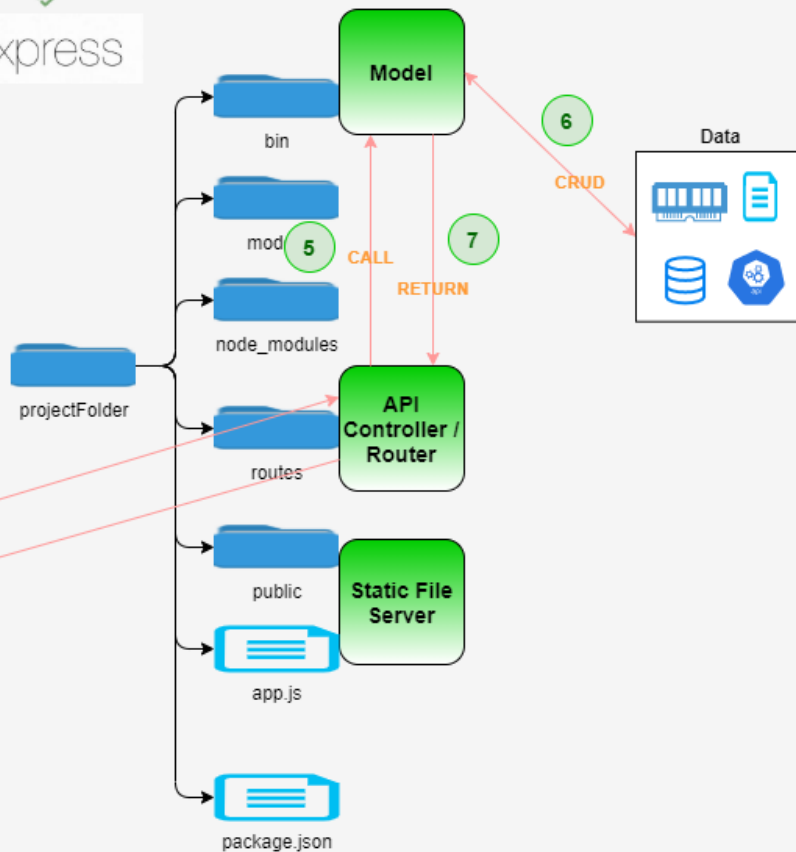
res.json(User.list)

Server



Express

Web application
& embedded web server





Routage des composants d'une SPA

Comment afficher le bon composant sur ma SPA ?

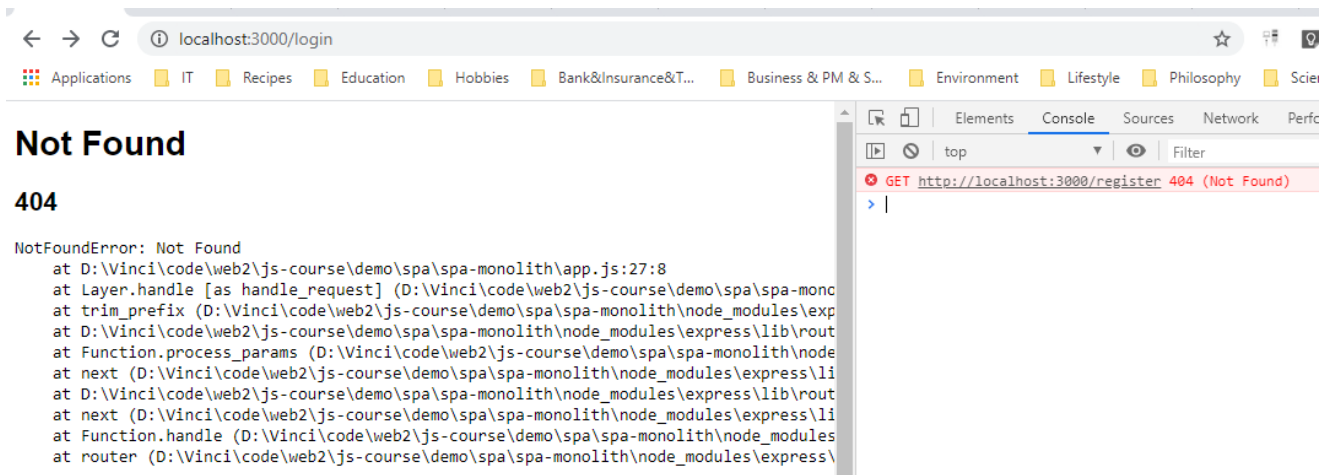


SPA : possibilités pour le routing de vos composants

- Créer un routeur en Vanilla JS :
 - Affichage du bon composant lors d'un clic sur la navbar
 - Affichage du bon composant au chargement de la page
 - Source d'inspiration [\[72.\]](#)
- Bibliothèques existantes : page.js, Vaadin Router...



SPA : quid si refresh de votre browser ?





SPA : résolution des problèmes de refresh

- Backend :
 - Renvoi d'**index.html** quelque soit la ressource demandée
 - Création d'un middleware
- Frontend : affichage du composant associé à l'URL donnée dans le browser au chargement de la page

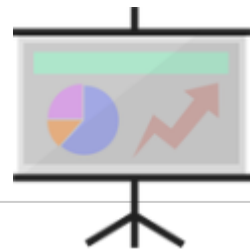


Routage des composants d'une SPA

🕒 DEMO : Routage des composants d'une SPA

Transformation de notre MPA MyCMS vers
une SPA statique :

- Création des composants LoginPage, HomePage... à l'aide de template literals ``
- Mise en place du router lors des clics sur la navBar
- Gestion du refresh



Routage des composants d'une SPA



🕒 DEMO : Routage des composants d'une SPA

- **data-*** attributes : en JS, conversion de - en camelCase



```
<a class="nav-item nav-link" href="#" data-uri="/home" data-extra-info="Home sweet home">Home</a>
```



```
const onNavigate = (e) => {  
  let uri;  
  uri = e.target.dataset.uri;  
  console.log("extra info associated to this link:", e.target.dataset.extraInfo,  
    // ... extra info associated to this link: Home sweet home  
  );  
  navbar.addEventListener("click", onNavigate);  
};
```