

VUE

VUE(CLI)

DURE TOTALE DU MODULE : 21H00

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Table des matières

Objectifs pédagogiques :	5
Présentation et installation de VUE	6
Présentation de VUE :	6
Approche Composants - Problème (CDN)	9
Approche Composants – fonction component (CDN)	11
Installation de VUE (mode CDN) :	13
Installation de VUE (mode CLI)	13
Présentation vue (mode cli)	16
Installation-configuration vue (mode cli)	17
Analyse des fichiers de base d'une app vue (mode cli)	21
Création d'une app vue (mode cli) from scratch	26
Exercice : Re créer une app VUE	27
Components Communication	29
Props (communication composant parent vers composant enfant)	29
Exercice : component communication (props)	30
Exercice : component communication (props) problème	32
Exercice : component communication (props) correction du problème	34
Exercice : component communication (props) plus détaillé	36
Props + v-for : composants dynamiques	38
Exercice : component communication (props + v-for) Composants dynamiques	39
\$emit() (communication composant enfant vers composant parent)	40

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (\$emit()) test :réagir à un évènement custom	41
Exercice : component communication (\$emit()) test :réagir à un évènement custom + données	42
\$emit() (plus en détails)	43
Exercice : component communication (\$emit()) en mode objet	44
Exercice : component communication (\$emit()) + nouveau composant	46
Exercice : component communication (\$emit()) + nouveau composant + suppression	47
TP : Component Communication	49
Component Registration : comment et où enregistrer les composants	50
Exercice : component Registration : test avec un composant	51
Exercice : component Registration : on déclare tous les composants dans App	52
Scoped Styles	53
Exercice :component Style Scoped	54
Slots : une autre manière de passer des données dans un composant	55
Exercice component slots	56
Exercice component multi slots	58
VUE + Firebase (communiquer avec une BDD)	60
setup de Firebase (console web)	60
SetUp de l'application VUE	67
Exercice : Setup de l'app	67
Router	68
Exercice : installation / configuration du router	69
Exercice : Utilisation du router (ajout navbar)	70

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Installation / configuration de firebase	71
Installer Firebase dans le projet	71
Exercice Config de la Connexion à Firebase	72
Exercice : Mise en place du Service (utilisation de class JS) via ajout en BDD	73
Cycle de vie des composants (LifeCycle Hooks)	76
Exercice : Liste des articles	78
Exercice :Création du composant article	82

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Objectifs pédagogiques :

Être capable de comprendre les enjeux des Frameworks

Être capable d'initialiser et paramétriser un projet grâce à un Frameworks front-end

Être capable de réaliser des interface utilisateur grâce à un Frameworks front-end

Être capable de mettre en place un système de Routing (url)

Être capable de gérer des requêtes HTTP (BDD)

Être capable de connecter une application à une base de données

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Présentation et installation de VUE

Présentation de VUE :



Qu'est-ce que VUE ?

VUE est un **Framework Javascript** permettant de construire des applications **Front End** interactives, et **réactives** plus facilement.

Framework Javascript : ensemble de librairie(s) tierce(s) qui met à disposition des utilitaires, des fonctionnalités et des bonnes pratiques sur comment construire des applications Javascript

Javascript : Un langage de programmation supporté par tous les navigateurs.
Permet de manipuler les éléments HTML, les données dans une page web.

Front End : « Ce que l'utilisateur voit » HTML + CSS + Javascript.

Réactives : L'App est capable de réagir aux actions de l'utilisateur et mettre à jour ce qui s'affiche à l'écran. ==> Le Look et le Feeling d'une appli mobile.

« À la différence des autres Frameworks monolithiques, Vue a été conçu et pensé pour pouvoir être adopté de manière incrémentale. Le cœur de la bibliothèque est concentré uniquement sur la partie vue, et il est vraiment simple de l'intégrer avec d'autres bibliothèques ou projets existants. D'un autre côté, Vue est tout à fait capable de faire tourner des applications web monopages quand il est couplé avec des outils modernes et des bibliothèques complémentaires. »

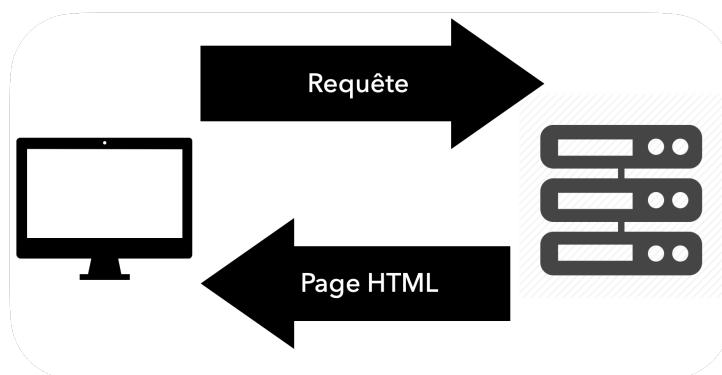
Vue.js a été créé à l'origine par une seule personne, Evan You, ancien ingénieur de Google ayant notamment travaillé avec Angular.js puis sur le framework Meteor. Après avoir expérimenté une mécanique de réactivité qu'il jugeait plus intéressante que celle d'Angular, il décide de publier ses

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

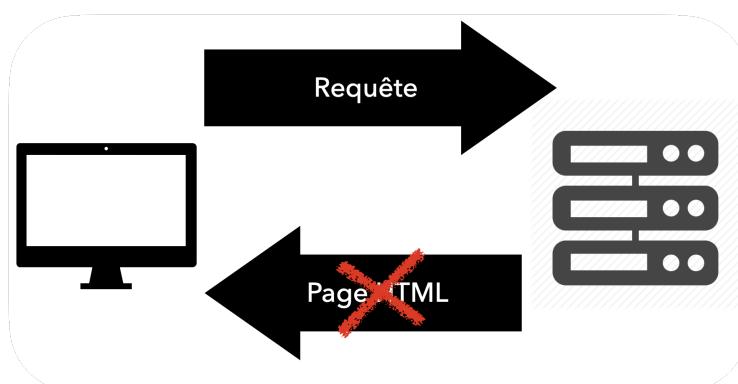
VUE

premiers résultats en juillet 2013. Cinq ans plus tard, Vue.js est le 3ème projet sur Github en nombre de stars et Evan enchaîne les conférences à travers le monde.

Généralement, dans des applications web « classiques », les interactions avec le serveur (click sur un bouton pour envoyer un message par exemple) entraînent un rechargeement de la page.



L'arrivée de Javascript a permis d'éviter ces rechargements de pages, dans le sens où JS s'exécute dans le navigateur sur la page qui est chargée, JS va pouvoir manipuler les éléments du DOM de la page.



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Sur l'aspect technique, il est tout à fait possible de construire n'importe quelle application web en se basant sur du Javascript, sans recourir à l'utilisation d'un Framework (Angular, Vue, React, Svelte ...)

Mais :

- Nous devrons écrire tout le code de l'application (On réinvente la roue ?)
- Risque de réaliser un code non optimisé comprenant des erreurs, des bugs (difficilement maintenable / scalable).
- Travailler en équipe sera plus difficile, tout le monde ne connaît pas votre manière de coder.

L'utilisation d'un Framework permet de répondre à ces problématiques, il propose un ensemble de composants réutilisables, un cadre de travail avec des règles à respecter ainsi que de bonnes pratiques.

VUE adopte une approche différente des autres frameworks (monolithique), Vue se définit comme un framework progressif, on peut choisir d'utiliser de deux manières :

L'approche plutôt en mode widget (maquette gauche) (Vue contrôle 1 ou plusieurs parties spécifiques d'une application), cela permet de facilement intégrer et utiliser vue dans des projets existants.

L'approche en mode « monolithique » (maquette droite), on gère toute une application en VUE



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Approche Composants - Problème (CDN)

Dans l'exemple suivant nous allons re créer une application qui va lister (des amis / contacts/ produits, au choix).

Nous allons développer cette mini application normalement (approche avec CDN).

JS :

Dans les data on a :

1 variable « detailsVisible » initialisée à false

1 variable lesAmis qui est un tableau qui va contenir 2 objets

Dans ces objets on a un id, name, phone, email (initialisé avec des données au choix)

Dans methods on a une f° toggleDetails dans laquelle à detailsVisible on assigne son inverse

HTML :

```
<div class="card border-dark mb-3">
    <div class="card-header"></div>
    <div class="card-body text-dark">
        <button type="button" class="btn btn-dark mb-1">MSG Button</button>
        <ul class="list-group list-group-flush">
            <li class="list-group-item"></li>
            <li class="list-group-item"></li>
        </ul>
    </div>
</div>
```

°Sur la div card mettre en place un v-for (variable temporaire : unAmi) pour parcourir le tableau lesAmis avec un v-bind de l'attribut key pour le lier avec unAmi.id

°Dans la div card-header : afficher unAmi.name

°Sur le bouton on réagit au click pour déclencher la methods toggleDetails

°Sur la mettre en place un v-if qui vérifie si la variable detailsVisible est définie

°Dans les on va afficher le mail et le phone de unAmi

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

 Les Copains D'Abord
COCO L ASTICOT
<input type="button" value="Masquer"/>
coco@lasticot.com
01234 5678 991
 Janine DeLavega
<input type="button" value="Masquer"/>
janine@delavega.com
09876 543 221

Avec cette approche le problème c'est que le bouton afficher / masquer Liste fonctionne pour les 2 profils.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Approche Composants – fonction component (CDN)

JS setup:

- ° Juste avant app.mount('#app');
- ° Sur app on va utiliser la f° component() mise à dispo par VUE
- ° la f° component() prend
- ° un 1er param 'ami-comp'
- ° Un 2nd param un objet JS
- ° Dans cet objet JS {} on va avoir :
- ° 1 propriété template (une string vide pour le moment ...) utiliser template strings ``
- ° 1 f° data() {} (vide pour le moment ...)
- ° 1 propriété methods (un objet vide pour le moment ...)

JS Modif avec fonction component:

- ° Dans app.component('ami-comp', {});
- ° Dans les data(){ } on retourne un objet dans lequel on a :
- ° 1 variable detailsVisibles à false (on peut commenter le detailsVisibles du Vue.createApp)
- ° 1 variable unPote c'est un objet qui contient un id, un name, un phone un mail (tout ça au choix)
- ° Dans methods on va recopier notre fonction toggleDetails (on peut commenter l'ancienne method toggleDetail du Vue.createApp)

HTML :

- ° Sélectionner toute la div class card (celle qui englobe tout), puis ctrl + c, et on peut mettre en commentaire
- ° Ce template HTML on va le placer dans la propriété template côté JS
- ° Modifier dans le template les variable UnAmi en unPote à la place
- ° dans app.component('ami-comp', { }) (On peut enlever la partie v-for sur la div class card)
- ° Retour côté HTML notre est un peu vide on va placer plusieurs balises HTML un peu custom : <ami-comp></ami-comp>

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

 [Les Copains D'Abord](#)

coco l'asticot

[Masquer](#)

coco@lastico.com

123 12346 24

coco l'asticot

[Voir](#)

Dans cette approche avec la fonction component nous avons créé un nouveau composant <un-ami>, qui possède ses propres data et ses propres methods.

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Installation de VUE (mode CDN) :

En mode CDN, nous n'allons pas Installer VUE, nous allons directement l'utiliser depuis son server d'hébergement.

<https://vuejs.org/guide/quick-start.html#without-build-tools>

Il suffira d'intégrer cette balise dans nos pages HTML :

```
<script src="https://unpkg.com/vue@3"></script>
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Installation de VUE (mode CLI)

(Nous mettrons en place cette technique dans la seconde partie du cours)

En mode CLI (Command Line Interface), nous pouvons créer des projets via des lignes de commande (terminal) entièrement gérés en VUE.

Cette méthode d'installation passe par des outils de gestion de package (il en existe plusieurs mais le plus commun est l'outil NPM qui fait partie de NodeJS)

NPM

<https://nodejs.org/en/>

Une fois NodeJS installé vous pouvez utiliser son outil de gestionnaire de package NPM pour installer dans vos projets tout type de librairies, Framework... et notamment VUE CLI

Dans un terminal, la commande ci-dessous va installer VUE CLI sur votre machine :

- npm install -g @vue/cli

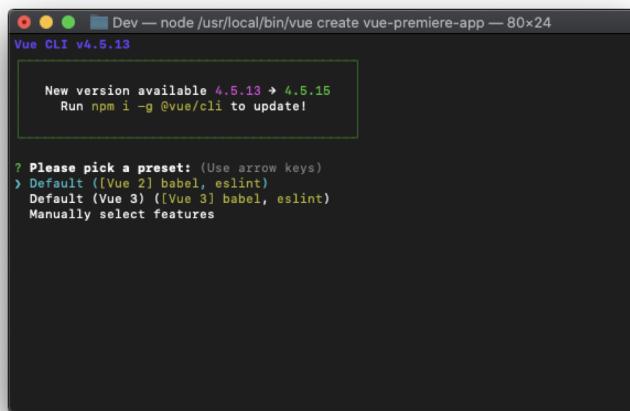
Le « -g » va permettre d'installer VUE CLI globalement sur votre machine vous pourrez par la suite utiliser des commandes pour créer des projets VUE dans n'importe quel dossier de votre machine.

Pour créer et initialiser un nouveau projet VUE :

- Se déplacer dans le terminal dans un dossier (avec la commande cd)
- Et créer une nouvelle application avec la commande :
vue create le-nom-d-application

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

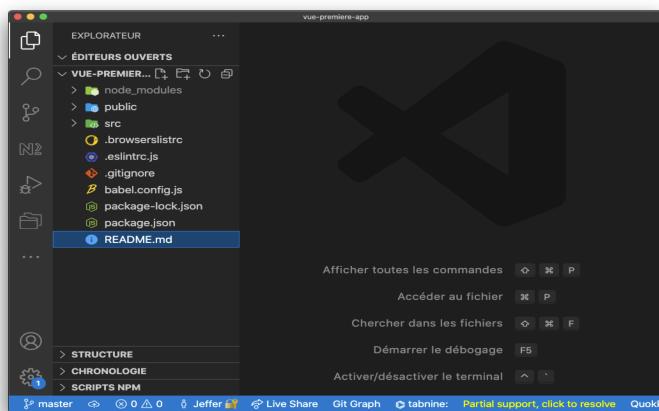


Le CLI de VUE permet de créer et initialiser directement tout un projet, nous pouvons choisir entre Vue2, Vue3 (La dernière version de Vue que nous utiliserons) et la possibilité de configurer manuellement un projet.

Le CLI va donc créer un nouveau dossier pour votre projet (que vous pouvez ouvrir dans VSCode) avec tout un ensemble de fichiers de base.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Dossier `node_modules` : c'est dans ce dossier que sont installées toutes les packages, dépendances du projet (exemple si l'on choisit d'installer Bootstrap dans notre projet, les fichiers de Bootstrap se retrouveront dans `node_modules`)

Dossier `public` : où l'on retrouve la page `index.html`, la seule page de l'application (qui affichera un composant).

Dossier `src` : c'est dans ce dossier que l'on développera nos composants, de base toute l'application vue possède un composant racine le fichier `App.vue`, ainsi qu'un autre composant `HelloWorld.vue`

Fichier `eslint` : le fichier de configuration de l'analyseur de code

Fichier `babel` : le fichier de configuration de Babel (un compilateur JS), son rôle est de traduire des fichiers `.vue` en JavaScript interprétable directement par les navigateurs

Fichier `package.json` : le fichier qui va lister les dépendances du projet.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

YARN

Une alternative à l'outil npm de node c'est un autre gestionnaire de package : yarn , son but est le même que NPM à savoir pouvoir facilement télécharger des dépendances

<https://classic.yarnpkg.com/en/docs/install>

On peut l'installer via npm :

```
npm install --global yarn
```

et on peut utiliser cette commande pour vérifier la version :

```
yarn --version
```

Ci-dessous vous pouvez trouver les principales commandes comparées à celle de NPM :

<https://classic.yarnpkg.com/en/docs/migrating-from-npm#toc-cli-commands-comparison>

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Vite

Une alternative au CLI classique dans VUE, il s'agit d'un outil également développé par les équipes de VUE qui permet les mêmes choses que le CLI de Vue mais qui va plus ... vite !

Vite permet de rapidement créer et initialiser des projets avec différents frameworks (Vue, React, Svelte..)

<https://vitejs.dev/guide/>

Dans un terminal nous pouvons utiliser ces commandes pour générer un projet :

npm create vite@latest

ou

yarn create vite

Le terminal nous demande un nom de projet

```
hkd@HKD ~ % cd /Users/hkd/Documents/GitHub
hkd@HKD GitHub % npm create vite@latest
? Project name: > vite-project
```

Une technologie pour le projet

```
? Project name: ... test-vite
? Select a framework: > - Use arrow-keys. Return to submit.
  Vanilla
  > Vue
  React
  Preact
  Lit
  Svelte
  Others
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans une technologie on nous propose le choix (pour VUE) entre Javascript et TypeScript :

```
hkd@HKD GitHub % npm create vite@latest
✓ Project name: ... test-vite
✓ Select a framework: > Vue
? Select a variant: > - Use arrow-keys. Return to submit.
> JavaScript
  TypeScript
  Customize with create-vue ↗
  Nuxt ↗
```

Puis VITE va nous créer le projet et tous les fichiers :

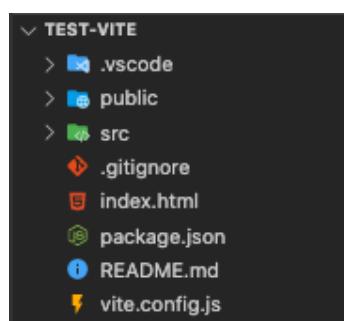
```
hkd@HKD GitHub % npm create vite@latest
✓ Project name: ... test-vite
✓ Select a framework: > Vue
✓ Select a variant: > JavaScript

Scaffolding project in /Users/hkd/Documents/GitHub/test-vite...

Done. Now run:

  cd test-vite
  npm install
  npm run dev

hkd@HKD GitHub %
```



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Présentation vue (mode cli)

Dans la première partie du cours, en mode cdn, nous avons pu voir les principales les directives de base que l'on utilise le plus souvent dans les applications web.

L'approche CDN a un côté pratique dans le sens où l'on peut facilement combiner le Framework Vue avec des applications existantes.

Pour des applications plus complexes (dans lesquelles on veut contrôler plus de chose via Vue) nous avons entrevu l'approche en mode composant.

Le Framework propose également de gérer toute une application web avec VUE, avec le CLI (Command Line Interface). Il s'agit de commandes à inscrire dans le terminal de votre ordinateur, ces commandes vont permettre entre autres de faciliter la création et le paramétrage d'un projet d'application.

Nous aurons un meilleur WorkFlow et les applications seront plus facilement scalable.

💡 Pourquoi avons-nous besoin d'un meilleur Workflow de développement / DX ?

Pour avoir à disposition un server de développement ce qui permet entre autres de recharger automatiquement vos page web à chaque modification du code source.

(certaines fonctionnalités de JS ES6 et + ne fonctionnent pas bien sans serveur (de développement ou de production)). De meilleures autocomplétions et messages d'erreur dans notre IDE.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Installation-configuration vue (mode cli)

⚠ ne pas confondre : VUE CLI est un outil en plus de VUE Framework (développé par la team VUE) pour mieux créer et gérer des projets.

<https://cli.vuejs.org/>

VUE CLI se sert d'autres outils, des Package Manager (des gestionnaires de paquets), pour résumer les paquets ce sont du code source dans des dossiers, on s'en sert souvent pour télécharger les fichiers sources de librairies tierces au sein de notre projet (ex : avec une commande dans le terminal on télécharge le code de Bootstrap dans notre application VUE)

Pour les exemples d'applications que nous allons créer, nous allons utiliser NPM (Node Package Manager) c'est l'un des Package Manager les plus populaire / utilisé.

Nous n'allons pas faire de développement NodeJS, nous allons juste utiliser son outil NPM.

<https://nodejs.org/en/>

Pour vérifier si node est correctement installé Taper cette commande dans le terminal :

« node -v » cela vous donnera un numéro de version.

(-g va permettre d'installer VUE CLI en global sur votre machine ,on pourra créer des projets Vue dans n'importe quel dossier /emplacement de l'ordinateur)

❗ Si vous rencontrez des problèmes lors de l'installation :

Sur Windows exécutez le terminal en tant qu'administrateur

Sur Linux / Mac rajouter le mot « sudo » avant de taper la commande

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

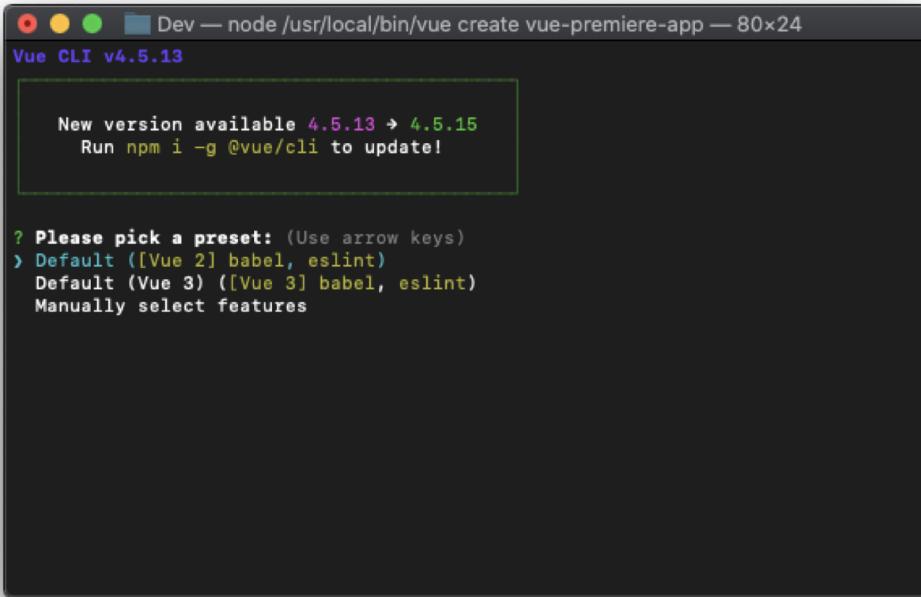
VUE

Toujours dans le terminal, nous allons nous déplacer dans un dossier (au choix : le bureau, dans un dossier dev, sur un disque dur, une clé usb, etc..)

<https://fr.wikihow.com/changer-de-r%C3%A9pertoire-dans-le-mode-de-commande>

Une fois dans le bon dossier toujours dans le terminal on peut faire la commande « vue create leNomDeLapplication », par exemple :

« vue create vue-premiere-app », le CLI de VUE va alors créer un dossier pour votre application avec plusieurs fichiers.



```
Dev — node /usr/local/bin/vue create vue-premiere-app — 80x24
Vue CLI v4.5.13

New version available 4.5.13 → 4.5.15
Run npm i -g @vue/cli to update!

? Please pick a preset: (Use arrow keys)
> Default ([Vue 2] babel, eslint)
Default (Vue 3) ([Vue 3] babel, eslint)
Manually select features
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Le CLI nous propose plusieurs choix VUE2, VUE3, ou une installation manuelle, nous allons sélectionner VUE3

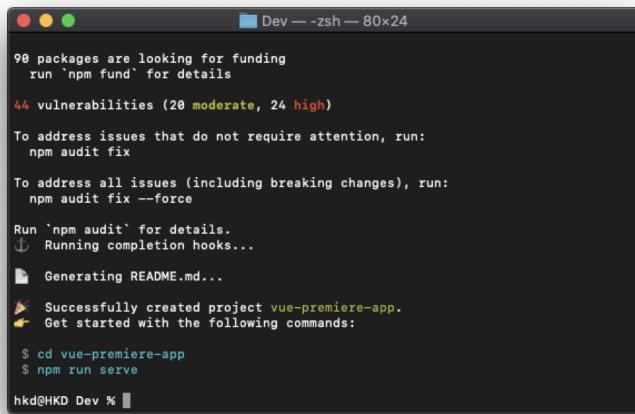
Le CLI va créer le dossier de votre application

```
● ○ ● Dev — npm -> node /usr/local/bin/vue create vue-premiere-app — 80x24
Vue CLI v4.5.13
  Creating project in /Users/hkd/Dev/vue-premiere-app.
  Initializing git repository...
  Installing CLI plugins. This might take a while...

(   ) " idealTree:vue-premiere-app: sill idealTree buildDeps
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE



```

Dev — zsh — 80x24
90 packages are looking for funding
  run 'npm fund' for details

44 vulnerabilities (20 moderate, 24 high)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
↳ Running completion hooks...

📄 Generating README.md...
📝 Successfully created project vue-premiere-app.
👉 Get started with the following commands:

$ cd vue-premiere-app
$ npm run serve

hkd@HKD Dev %

```

Ensute on va ouvrir ce dossier crée par le CLI dans VSCode, puis dans le terminal lancer la commande « `npm run serve` » cela va compiler l'application et l'exécuter sur un serveur de développement en local.

Pour accéder à l'application sur le serveur (8080 est le port par défaut mais on peut le modifier si besoin) :

<http://localhost:8080/>

Nous avons une application VUE (de base), pré-configurée.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Cool : Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project,
check out the [vue-cli documentation](#).

Installed CLI Plugins

[babel](#) [eslint](#)

Essential Links

[Core Docs](#) [Forum](#) [Community Chat](#) [Twitter](#) [News](#)

Ecosystem

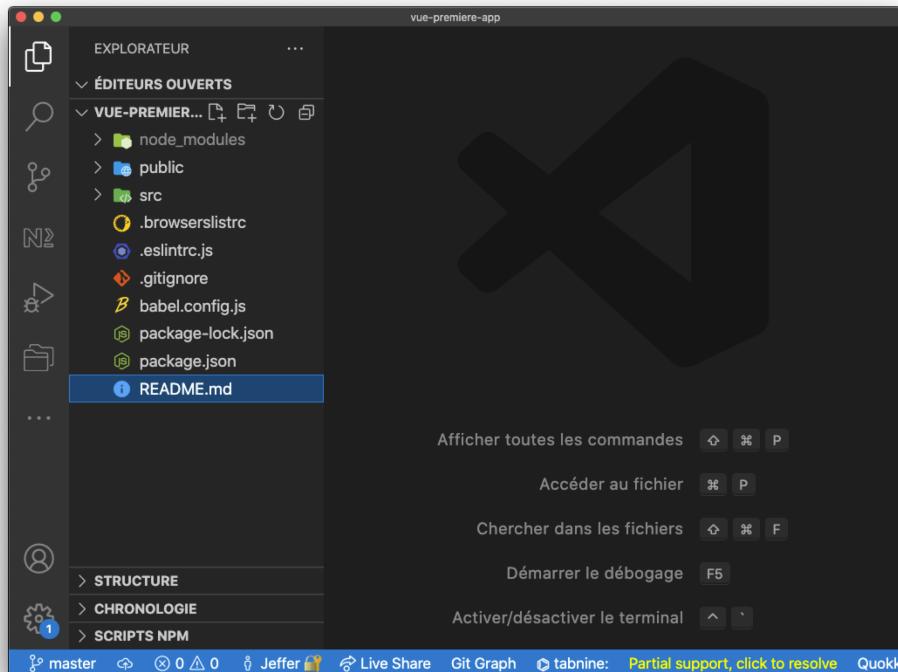
[vue-router](#) [vuex](#) [vue-devtools](#) [vue-loader](#) [awesome-vue](#)

Analyse des fichiers de base d'une app vue (mode cli)

On se retrouve donc non plus avec juste un fichier HTML et un fichier JS, mais avec plusieurs fichiers de configuration et des dossiers.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

À la racine du projet nous retrouvons :

Des fichiers de configuration :

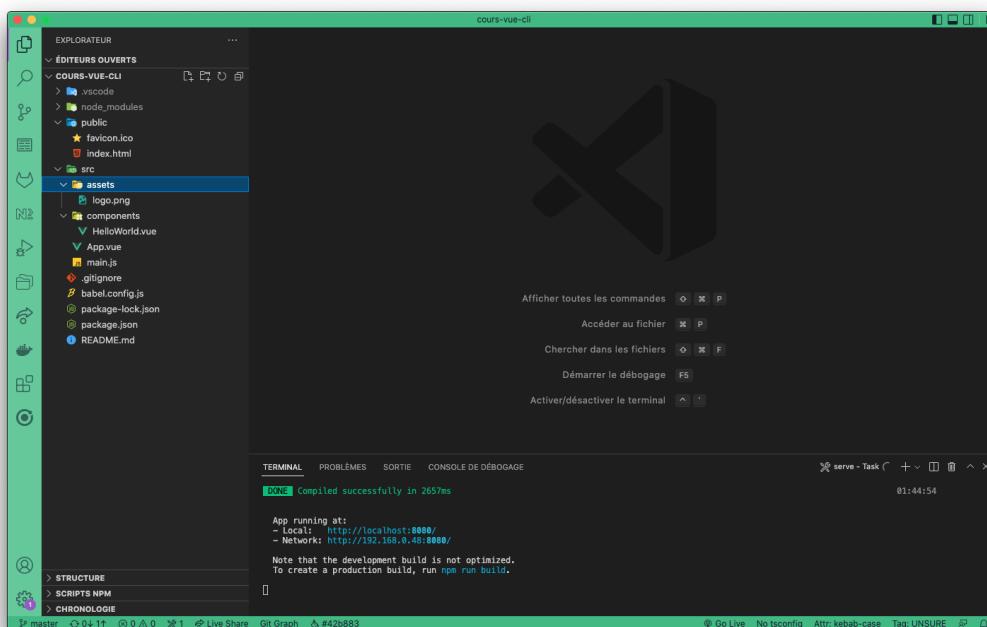
- Babel (compiler JS)
- eslint (l'analyseur de code)
- package.json (liste des dépendances à des package)

1 dossier node modules (l'endroit où vont s'installer les package / dépendances du projet)

ex : si j'installe Bootstrap dans le projet cela va télécharger les fichiers de Bootstrap dans node modules et mettre à jour package.json

1 dossier public qui contient un fichier index.html (notre SPA toute l'appli sera dans index)

1 dossier src (tout le code source de l'application)



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Le fichier index.html (dans le dossier public), on retrouve une div avec l'id « app », toute l'application vue est dans cette div.

```
<!DOCTYPE html>
<html lang="">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="<%= BASE_URL %>favicon.ico">
    <title><%= htmlWebpackPlugin.options.title %></title>
  </head>
  <body>
    <noscript>
      <strong>We're sorry but <%= htmlWebpackPlugin.options.title %> doesn't work
properly without JavaScript enabled. Please enable it to continue.</strong>
    </noscript>
    <div id="app"></div>
    <!-- built files will be auto injected -->
  </body>
</html>
```

Le Fichier main.js, on retrouve des éléments que l'on connaît

```
import { createApp } from 'vue'
import App from './App.vue'

createApp(App).mount('#app')
```

L'import d'une fonction createApp et d'une variable App, l'utilisation de la fonction mount() avec l'id de la div côté HTML.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans le dossier components, nous avons des fichier en .vue, un composant VUE c'est juste une combinaison dans le même fichier de HTML / JS / CSS (le template / la Logique / le Style)

 <https://marketplace.visualstudio.com/items?itemName=octref.vetur>

(une extension VSCode pour que l'IDE comprenne mieux les fichiers .vue)

App.vue :

```
<template>
  
  <HelloWorld msg="Cool : Welcome to Your Vue.js App"/>
</template>

<script>
import HelloWorld from './components/HelloWorld.vue'

export default {
  name: 'App',
  components: {
    HelloWorld
  }
}
</script>

<style>
#app {
  font-family: Avenir, Helvetica, Arial, sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  text-align: center;
  color: #2c3e50;
  margin-top: 60px;
}
</style>
```

Dans le fichier App.vue nous pouvons remarquer une balise HTML étrange

```
<HelloWorld msg="Cool : Welcome to Your Vue.js App"/>
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

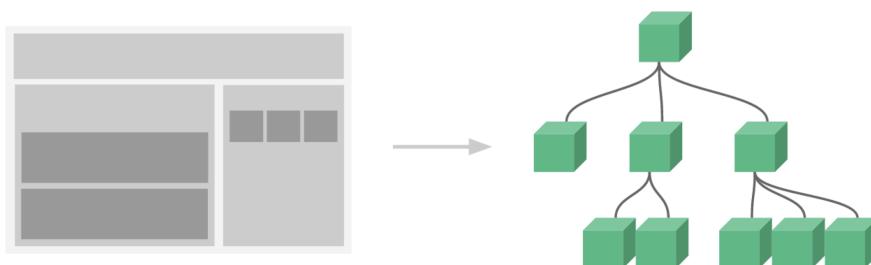
Dans le script on importe et exporte le composant HelloWorld

```
import HelloWorld from './components/HelloWorld.vue'

export default {
  name: 'App',
  components: {
    HelloWorld
  }
}
```

En fait App.vue, ou le composant App est le composant principal de toute application VUE (le root component ou composant racine)

Schéma : Organisation classique des composants d'une application VUE, on a le App component dans lequel sont imbriqués d'autres composants. (Chaque partie d'un site, (header, nav, section,etc..) peuvent être un composant)



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Création d'une app vue (mode cli) from scratch

Pour mieux comprendre la notion et l'utilisation des composants nous allons dans un premier temps créer le composant principal d'une application VUE par la suite nous allons créer un second composant.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : Re créer une app VUE

Récupérer blankProject

Ouvrir le projet dans VScode puis installer les dépendances avec la commande terminal : « npm i » (raccourci de « npm install »), ensuite taper la commande « npm run serve »

Dans le dossier « src » créer un fichier App.vue

- ° Dans App.vue, créer une balise <template> puis une balise <script>
- ° Dans la balise <template>, on a juste un titre et une balise
- ° Dans la balise <script>, on va exporter par défaut un objet comme ceci : export default {}
- ° Dans cet objet on a une fonction data() {}
- ° Dans data(), on retourne un objet
- ° Dans l'objet on copie colle le tableau « lesAmis » (de l'app freindList)

Dans le fichier main.js (dans le dossier « src »)

- ° À la ligne 2 on import App from './App.vue'
- ° Dans la fonction mount() on lui passe « #app » (l'id de la balise dans index.html)
- ° On passe la variable App en paramètre de createApp()

Dans le dossier « src » créer un dossier « components »

- ° Dans le dossier « components », créer un fichier « UnAmi.vue »
- ° Dans **UnAmi.vue** : il nous faut un <template> et un <script>
- ° Dans le <script>, on fait export default {}, (ancien projet), une variable « detailsVisible » à false et un objet «unPote » (avec un id,name,phone et email), on a aussi la methods afficherDetails qui assigne à « detailsVisible » son inverse
- ° Dans le <template>, dans une liste, on affiche le name de unPote, un bouton qui déclenche la fonction « afficherDetails », un autre liste sur laquelle on place un v-if qui vérifie « detailsVisible », dans cette dernière , des doivent afficher le phone et l'email de unPote

Dans le fichier main.js (dans le dossier « src »)

- ° Juste après l'import de « App », on va importer UnAmi from './components/UnAmi'
- ° mettre en commentaire cette ligne : createApp(App).mount('#app')
- ° on va plutôt faire comme ça : const app = createApp(App); puis app.mount('#app')
- ° ensuite sur « app » on va utiliser la fonction component() en lui passant 2 paramètre :
 - ° une string : le nom sous lequel on veut utiliser ce composant ('un-ami')
 - ° la variable UnAmi (que l'on a importé plus haut)

Dans le fichier App.vue

- ° Dans la liste on va placer 2 fois la balise <un-ami>

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Les Amis

jojoLeBarjo

[Voir Détails](#)

123 12346 24

jojo@barjo.com

jojoLeBarjo

[Voir Détails](#)

[GitHub](#)

Cet exercice nous a permis de recréer une application de base VUEJS, en mettant en exergue l'interaction entre les différents fichiers.

Par la suite nous utiliserons la base d'application construite par le CLI.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Components Communication

Props (communication composant parent vers composant enfant)

Dans l'application précédente nous avions recréé le setup de base d'une application VUE, nous avons créé le ROOT Component App.vue (le composant principal/racine de toutes les application VUE), et un second composant qui se charge d'afficher un contact (UnAmi.vue), le problème c'est que l'on se retrouve avec 2 fois le même composant avec 2 fois les mêmes data.

L'idée serait plutôt de pouvoir utiliser ce composant en lui injectant des données.

Ce concept d'imbrication de composant est au cœur de la conception des applications web SPA (des application « component oriented »)

Le système de composant améliore la flexibilité lors du développement d'une app (on peut réutiliser facilement un composant dans plusieurs parties de l'application).

Nous allons donc maintenant nous intéresser à comment faire communiquer des composants.

Avec ce système de composants imbriqués, VUE dispose d'un concept pour faire communiquer un composant « parent » et son composant Child (enfant).

Il s'agit de **props**, un TABLEAU dans lequel :

On va définir les propriétés du composant enfant que l'on rend accessible au composant parent

On va pouvoir utiliser des attributs HTML « custom » pour injecter des données dans un composant enfant.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (props)

Dans UnAmi.vue :

- ° Dans l'export default, rajouter une propriété props, un tableau qui va contenir plusieurs chaînes de caractères : « leNom », « lePhone » et « leMail »
- ° Pensez à modifier les données que l'on affiche dans le Template, avec l'interpolation désormais on va afficher les **props**

Dans App.vue :

- ° Sur nos balises <un-ami>, on va rajouter des attributs HTML qui ont strictement le même nom que les **props** en leur attribuant des valeurs au choix.

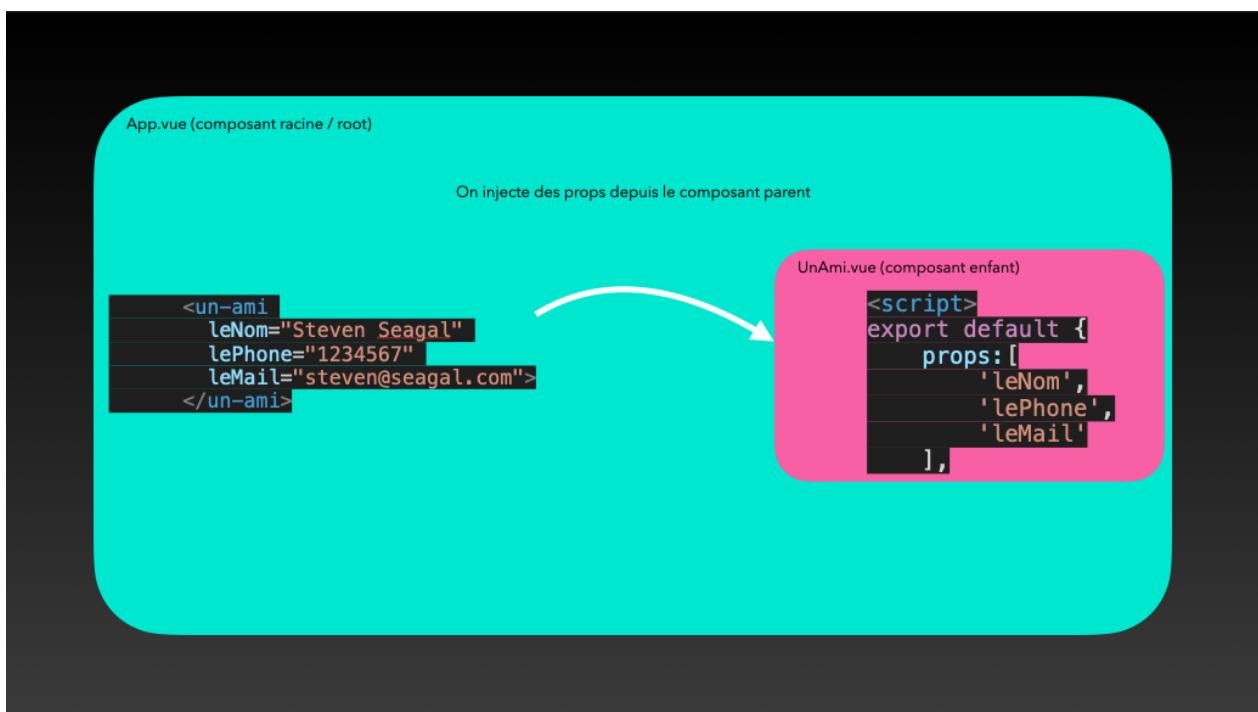
Les Amis

Steven Seagal
Voir Détails
1234567
steven@seagal.com
Gérard Menvuça
Voir Détails
0987654
gérard@menvuca.com

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (props) problème

On va tout casser !

Dans UnAmi.vue :

- ° Au niveau des props, rajouter une props « premium » (pour savoir si on a des ami de qualité 😊)
- ° Dans methods, rajouter une méthode « afficherPremium » dans cette méthode :
 - ° Si premium est bien égal à la chaîne de caractère « 1 », alors on met la props premium à « 0 », SINON on met la props premium à « 1 »
- ° Dans le template, au niveau du <h2>, on va rajouter une interpolation {{}}, dans laquelle on aura une condition ternaire : Si premium est bien égal à « 1 » on retourne « Ami Premium » sinon on retourne « Ami Nul »

Dans App.vue :

- ° Sur les balises <un-ami> on rajoute un attribut premium en injectant soit « 1 » ou « 0 ».

Yes 👍 :

```
Les Amis
Failed to compile.

./src/components/UnAmi.vue
Module Error (from ./node_modules/eslint-loader/index.js):
  Voir Détails
/Users/hkd/Documents/GitHub/cours-vue-
cli/src/components/UnAmi.vue
  39:17  error  Unexpected mutation of "premium" prop
  vue/no-mutating-props
  42:17  error  Unexpected mutation of "premium" prop
  vue/no-mutating-props

* 2 problems (2 errors, 0 warnings)
```

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Explication du « Bug » :

Pour la communication entre des composants avec props, Vue se base sur le concept du Unidirectionnal DataFlow (flux de données unidirectionnel), principalement pour des raisons de sécurité, de performances et de maintenance (debugging).

Dans notre cas les props que l'on injecte depuis App.vue vers UnAmi.vue ne devraient être modifiée QUE dans App.vue.

En revanche nous, dans UnAmi.vue, on essaie de modifier la props premium :

```
afficherPremium(){
    if(this.premium ==='1'){
        this.premium = '0';
    }
    else{
        this.premium = '1';
    }
}
```

Nous allons à l'encontre de ce flux de données **UNIDIRECTIONNEL**, c'est pour cela que nous avons ce type d'erreur :

error Unexpected mutation of "premium" prop vue/no-mutating-props

Pour corriger ce problème il faudrait que le composant parent soit informé que nous voulons modifier les **props** dans l'enfant. De cette manière le parent pourra mettre à jour ses data (plus tard).

Une autre manière de corriger cela serait de créer au sein du composant enfant (UnAmi.vue), une **data** liée à notre **props** premium.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (props) correction du problème

Dans UnAmi.vue :

- ° Au niveau des data, rajouter une data « premiumData » à laquelle on assigne la **props** premium
- ° Dans la methods « afficherPremium », remplacer les props « premium » par les data « premiumData »
- ° Dans le template, dans l'interpolation, remplacer les props « premium » par les data « premiumData »
- ° Dans le template, sous le <h2> rajouter un bouton qui réagit au click pour exécuter la methods afficherPremium

Les Amis

Steven Seagal Ami premium

Premium ?

Masquer

1234567

steven@seagal.com

Gérard Menvuça Ami nul

Premium ?

Masquer

0987654

gérard@menvuca.com

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Nous avons vu la configuration de base pour utiliser les props :

```
props: [
  'leNom',
  'lePhone',
  'leMail',
  'premium'
],
```

Avec cette configuration de base on sait que côté HTML on va injecter des données correspondantes à leNom, lePhone, leMail, premium (pas plus, pas moins).

Cette configuration est largement suffisante dans des petites applications, mais quand ça devient complexe ou que l'on travaille en équipe on va vouloir partager + d'informations sur les **props** pour que VUE puisse savoir si l'on injecte de mauvaises données ou non, ou si l'on a oublié d'injecter une **props** par exemple.

VUE permet d'aller plus en détails avec l'utilisation des **props**, on modifiant légèrement sa syntaxe, **props** serait désormais un objet { }, plutôt qu'un tableau [], on pourrait alors rajouter le typage des props et ou indiquer si la **props** est obligatoire.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (props) plus détaillé

Dans UnAmi.vue :

° Mettre en commentaire tout le tableau **props**, on va créer un objet **props** plutôt.

° Dans l'objet **props**, on va rajouter plusieurs objets (1 objet pour chaque **props** technique) :

```
  °leNom :{}
  °lePhone :{}
  °leMail :{}
  °premium :{}
```

° Dans chacun de ces objets on va avoir une propriété **type:String** (toujours Majuscule pour le type) et propriété **required** à true (sauf premium sa propriété **required** est à false)

° pour l'objet de la **props** premium, on rajoute une propriété **default** à '0', et une propriété **validator**

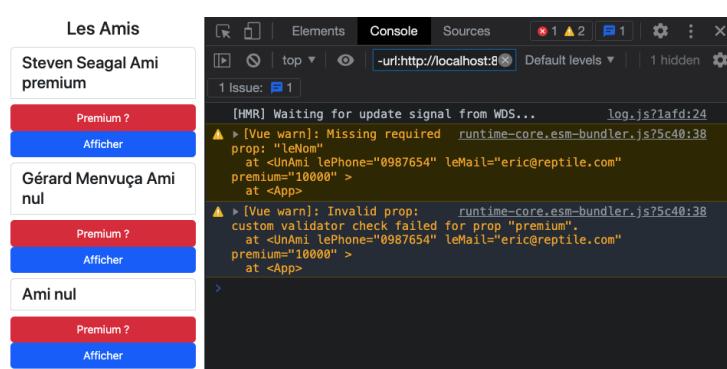
° à la propriété **validator** on lui attribue une fonction qui a une variable **value** en paramètre et qui **retourne** : cette condition value==='1' || value==='0'

Dans App.vue :

° Rajouter un composant <un-ami>, et nous allons créer des erreurs en injectant des props erronées par exemple :

° On oubli de renseigner la **props** leNom, et dans la **props** premium on va passer une mauvaise valeur « 1000 » par exemple.

° Ensuite dans le navigateur on va explorer la console pour voir les erreurs qui sont générés par cette nouvelle approche des **props** (plus sécurisé)



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

[GitHub](#)

Pour résumer avec l'approche objet des props :

```
props:{  
    premium:{  
        type:String,  
        required:false,  
        default:'0',  
        validator: function(value){return value==='1' || value==='0'}  
    }  
}
```

Nous pouvons gérer le **type** des données qui seront injectées dans le composant enfant, les rendre obligatoire ou non avec **required**, mettre une valeur par **default**(on peut utiliser également une fonction dans default), enfin les **validator** permette d'avoir des contrôles très précis et customisé avec des conditions spécifiques (ici par exemple le **validator** n'autorise que la chaîne de caractère « 1 » ou « 0 »).

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Props + v-for : composants dynamiques

A ce stade nous savons faire communiquer (injecter) des données d'un composant vers son composant enfant... Mais nous connaissons aussi la directive v-for... 😊

Afin de rendre notre composant encore plus dynamique et plus facilement réutilisable, dans le sens ou cela devient répétitif d'injecter une par une les props depuis le composant parent comme ceci :

```
<un-ami leNom="Gérard Menvuça" lePhone="0987654" leMail="gérard@menvuca.com" premium="0"></un-ami>
```

Dans l'exercice suivant nous allons gérer toutes les données dans les data de App.vue (composant parent) et optimiser leurs injections dans les **props** du composant enfant (UnAmi.vue).

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (props + v-for) Composants dynamiques

Dans UnAmi.vue :

- ° on va modifier l'objet de la **props** premium (le **type** sera booléen, le **required** et **default** seront à false)
- on peut mettre en commentaire le **validator** (plus besoin si c'est de type booléen)
- ° Au niveau des **data** on peut mettre en commentaire l'objet unPote
- ° Au niveau des methods, mettre en commentaire (ou supprimer) la partie IF ELSE, on va faire plus simple : on assigne premiumData à son inverse.
- ° Dans le template au niveau du <h2> modifier la condition ternaire : on vérifie juste si premiumData est définie (a voir on l'a ptet déjà fait 😊 cf. exo d'avant)

Dans App.vue :

- ° Mettre en commentaire (ou supprimer) les composants <un-ami> (on va en utiliser 1 seul).
- ° Sur un composant <un-ami> :
 - ° utiliser une directive v-for : avec une variable temporaire unAmi, parcourir le tableau lesAmis.
 - ° Ensuite on va faire des v-bind de :
 - key pour le lier à l'id de unAmi
 - leNom pour le lier à name de unAmi
 - lePhone pour le lier à phone de unAmi
 - leMail pour le lier à email de unAmi
 - premium pour lui assigner true

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Les Amis

COCO L ASTICOT (Ami premium)

Premium ?

Masquer

01234 5678 991
coco@lasticot.com

Janine DeLavega (Ami nul)

Premium ?

Afficher

[GitHub](#)

\$emit() (communication composant enfant vers composant parent)

Nous avons vu la notion de **props** dans VUEJS, cela nous permettait de transmettre des informations (des données) depuis un composant parent vers son composant enfant, nous avons également constaté que pour établir cette communication de données entre les composants, VUE se base sur un flux de données UNIDIRECTIONNEL (**unidirectional dataflow**), c'est-à-dire que pour des raisons d'optimisation / sécurité / etc..., on ne pouvait pas directement faire des modifications d'une **props** (une données injectée) dans le composant enfant (pour résoudre ce problème nous avions assigné une **data** à la **props** que l'on voudrait modifier).

Rappel : dans UnAmi .vue, pour pouvoir modifier la **props** premium dans ce composant on était passé par une **data** à laquelle on assigne la props.

```
export default {
  props: {
    premium: {
      type: Boolean,
      required: false,
      default: false
    }
  },
  data(){
    return{
      premiumData: this.premium
    }
  },
}
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans VUE c'est une approche au cœur du système de composant (et leur imbrication), sur des applications SPA (Single Page Application).

🤔 Quand on y pense :

Généralement c'est le root composant (racine) soit App.vue, qui va héberger / faire appel à des composants enfant (UnAmi.vue).

Mais quand on commence à complexifier les applications (gérant beaucoup de données, de fonctions, etc...), nous allons vouloir communiquer dans le sens inverse , que le composant enfant communique des informations au composant parent. (C'est du **unidirectional dataflow** mais dans l'autre sens).

Pour faire en sorte qu'un composant enfant envoie une information vers le composant parent, le composant enfant va devoir émettre un évènement (le changement d'une variable dans par exemple), on peut faire le parallèle où en JS on capte un évènement ('click', 'keyup', 'mouseout', ...) qui contient des informations (les coordonnées du click, la lettre tapée au clavier, etc..), ici c'est le même principe mais on crée des évènements « custom » qui correspondent aux besoins précis des données de l'application.

Dans le composant enfant cette communication se fait via une fonction de VUE :

`$emit('leNomDuEvent')`

Et dans le composant **parent** on va réagir à cet évènement :

`@leNomdeEvent = " du code JS / une variable / une fonction / etc..."`.

Dans l'exercice suivant nous allons faire en sorte que le composant enfant émette un évènement très simple (qui ne contient pas d'informations (pour le moment)) et dans le composant parent on va réagir à cet évènement en exécutant un console.log.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (\$emit()) test :réagir à un évènement custom

Dans unAmi.vue :

- ° Au niveau des **methods**, modifier la méthode afficherPremium, mettre en commentaire la manipulation de la variable premiumData, à la place on va utiliser this.\$emit() avec une chaîne de caractère en paramètre « mon-event-premium » (c'est le nom de l'évènement que le composant enfant va ... émettre).

Dans App.vue :

- ° Au niveau des **data**, pour chaque objet du tableau lesAmis, rajouter une propriété premium (à true ou false).
- ° Au niveau des **methods**, une méthode afficherStatusPremium dans laquelle on fait juste un console log (message au choix).
- ° Au niveau du template sur le composant <un-ami> :
 - ° modifier le **v-bind** de premium pour le lier à la propriété premium de unAmi.
 - ° mettre en place un **v-on** qui réagit à notre évènement « mon-event-premium » et appelle la méthode afficherStatusPremium



[GitHub](#)

On vient d'utiliser la fonction \$emit(), en passant 1 paramètre : le nom de l'évènement ...

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (\$emit()) test :réagir à un évènement custom + données

Dans UnAmi.vue :

- ° rajouter une props id (de type String, et required)
- ° techniquement on utilise plus premiumData (on peut mettre en commentaire dans les **data**), et dans la condition du <h2> (dans le template) utiliser premium (la **props** directement)
- ° Dans les **methods**, rajouter cette props id en second paramètre de \$emit()

Le composant enfant émet un event qui contient un id, on va le capter dans le composant parent

Dans App.vue :

- °Dans le template sur le composant <un-ami>, on peut rajouter un v-bind de id pour le lier à unAmi.id
- ° modifier la methods afficherStatusPremium pour qu'elle prenne une variable en paramètre (leIdDansUnAmi).
- ° Dans afficherStatusPremium :
 - ° créer une variable unAmiIdentified, lui assigner le tableau lesAmis (**data**), sur lesAmis utiliser la fonction. find(unTruc => unTruc.id === leIdDansUnAmi)
 - ° à la propriété premium de unAmiIdentified, lui assigner son inverse

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Les Amis

Captain Haddock (Ami nul)
Premium ?
Afficher
Steven Seagal (Ami premium)
Premium ?
Afficher

[GitHub](#)

.\$emit() (plus en détails)

Nous venons de voir une première manière d'utiliser la fonction emit(), mais pareillement à props, on peut aller plus loin dans la gestion des évènements avec \$emit() en passant par un tableau et même un objet, nous allons voir la version objet pour rajouter dans notre évènement, une fonction qui va vérifier si l'id est bien transmit.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (\$emit()) en mode objet

Dans UnAmi.vue :

- ° Dans l'export default, après les props, déclarer emits , un objet { }
- ° Dans cet objet on a une chaîne de caractère ('mon-event-premium') à laquelle on assigne une fonction (anonyme ou fléchée) qui prend id en paramètre.
- ° Dans cette fonction on va faire ceci :
- SI id est définie ALORS on return true SINON on fait un console.warn('MSG au choix') et on return false
- ° Pour tester (voir si le warning se déclenche), dans les methods retirer le paramètre this.id dans la fonction \$emit
- ° Pensez à le remettre par la suite.

```
⚠ ▶ ON A PAS DE ID                                         UnAmi.vue?47e4:53
⚠ ▶ [Vue warn]: Invalid event arguments: event validation failed for event "mon- runtime-core.esm-bundler.js?5c40:38
event-premium".
⚠ ▶ [Vue warn]: Unhandled error during execution of component event handler      runtime-core.esm-bundler.js?5c40:38
  at <UnAmi key="kimonoSurUnFrigo" id="kimonoSurUnFrigo" leNom="Steven Seagal" ... >
  at <App>
⚠ ▶ [Vue warn]: Unhandled error during execution of native event handler      runtime-core.esm-bundler.js?5c40:38
  at <UnAmi key="kimonoSurUnFrigo" id="kimonoSurUnFrigo" leNom="Steven Seagal" ... >
  at <App>
```

Cette approche permet de renforcer la sécurité des données quand elles communiquent entre composant en passant par des évènements.

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Schéma communication composants



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component communication (\$emit()) + nouveau composant

Nous allons rajouter un composant permettant d'ajouter un ami

Dans le dossier components :

- °Créer un nouveau fichier « NewAmi.vue »

Dans main.js :

- °importer le composant NewAmi, puis déclarer ce composant sous le nom « new-ami » dans l'app

Dans NewAmi.vue :

- ° Dans le **template** : 1 formulaire avec 3 inputs (pour ajouter un nom, un tél, un mail), et un bouton

- ° Pour la partie **script** juste la balise <script>: avec l'export default pour le moment

- ° Dans le **script**, dans export default, rajouter un tableau **emits** qui contient une chaîne de caractère « add-ami » (c'est le nom de l'évènement)

- ° Rajouter des **data** (des string vides) enteredName, enteredPhone, enteredMail

- ° Rajouter une **methods** submitData dans laquelle :

- ° on utilise la fonction **\$emit** en lui passant en paramètre « add-ami » (le nom de l'évènement) puis les **data** enteredName/Phone/Mail

Dans le template :

placer un v-model sur chaque input pour les relier aux **data**

sur la balise <form> placer un v-on qui écoute l'évènement « submit » (ajouter le **modifiers prevent**) et qui déclenche notre **methods** submitData

Dans App.vue :

- ° sur un composant <new-ami> placer un v-on qui écoute notre évènement add-ami et qui déclenche une **methods** ajouterAmi

- ° Rajouter une **methods** ajouterAmi qui prend en param name, phone et email.

- ° Dans cette fonction ajouterAmi, créer une variable newAmiContact, lui assigner un objet.

- ° Dans cet **objet** on a :

- une propriété id, lui assigner new Date().toISOString(),

- des propriétés name, phone et email, leur assigner name, phone et email,

- une propriété premium, lui assigner false

- ° Dans la data lesAmis, faire un push() de newAmiContact

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Les Amis

Name	TEST
Télé	0987654
Email	test@a.com
Ajouter Ami	
Captain Haddock (Ami premium)	
Premium ?	<input type="checkbox"/>
Afficher	
Steven Seagal (Ami nul)	
Premium ?	<input type="checkbox"/>
Afficher	
TEST (Ami nul)	
Premium ?	<input type="checkbox"/>
Afficher	

Exercice : component communication (\$emit()) + nouveau composant + suppression

Dans unAmi.vue :

- ° Dans le **template**, Sous la rajouter un bouton avec un **v-on** réagir au click et exécuter la fonction **\$emit()**, en lui passant 'delete' et id en paramètre
- ° Dans le script, mettre en commentaire l'objet emits (plus besoin maintenant qu'on a automatisé l'injection des données, le v-for sur <un-ami> dans App.vue)
- ° A la place on va juste passer par **emits** en mode **tableau** (qui contient les nom des event 'mon-event-premium' et 'delete').

Dans App.vue :

- ° Dans le **template**, sur le composant <un-ami> rajouter un **v-on** qui écoute l'event 'delete' et qui déclenche une methods effacerAmi
- ° Dans le script, rajouter une methods effacerAmi qui a unId en paramètre
- ° Dans effacerAmi on assigne au tableau lesAmis, lesAmis.filter(), c'est une fonction native de JS
- ° Dans les paramètres de filter(), on passe une fonction fléchée ()=>, (unePersonne)=> unePersonne.id !== unId

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

[GitHub](#)

Les Amis

Name			
aaa	<input type="button" value=""/>		
Tél			
aaaa	<input type="button" value=""/>		
Email			
a@a.com	<input type="button" value=""/>		
<input type="button" value="Ajouter Ami"/>			
CaptainHaddock (Ami nul)			
<input type="button" value="Afficher Détails"/>		<input type="button" value="Premium"/>	<input type="button" value="Suppr."/>
Steven Seagal (Ami nul)			
<input type="button" value="Afficher Détails"/>		<input type="button" value="Premium"/>	<input type="button" value="Suppr."/>
aaa (Ami nul)			
<input type="button" value="Afficher Détails"/>		<input type="button" value="Premium"/>	<input type="button" value="Suppr."/>

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

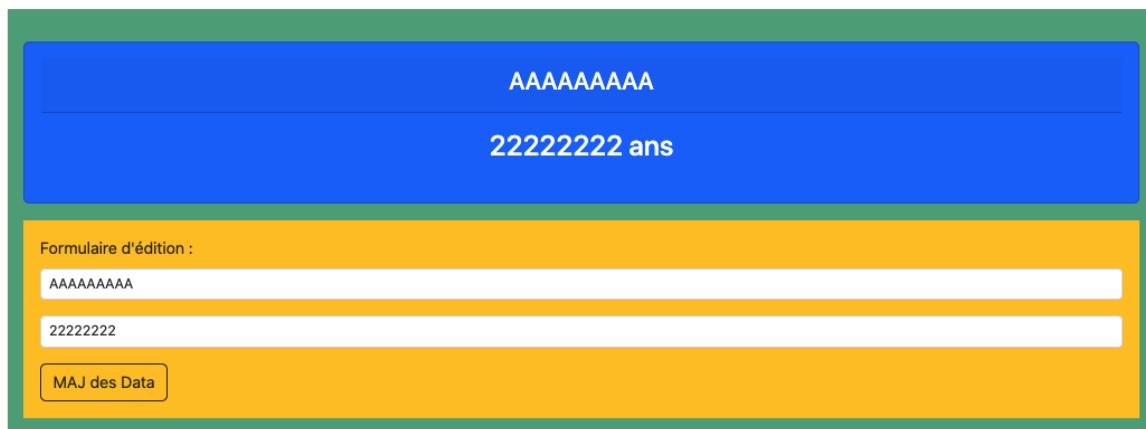
VUE

TP : Component Communication

A rendre Zippé en MP

Créer une Nouvelle application VUE dans laquelle :

- Créer 2 composants activeUser et userData
- Le composant activeUser affiche des infos de l'utilisateur (nom, age, image (au choix))
- Le composant userData comportera des inputs pour le nom et l'âge et un bouton « MAJ des données »
- Afficher ces composant dans App
- On doit pouvoir rentrer des data dans les inputs (s'assurer qu'on a bien au moins le nom et l'âge et le bon type de data)
- En cliquant sur le bouton « Maj des données » cela met à jours les données du user dans le composant activeUser



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Component Registration : comment et où enregistrer les composants

Pour le moment on enregistre tous nos composant dans main.js via la fonction `.component()`, c'est la méthode classique de **Registration** des composants mais quand ces derniers vont se multiplier, nous serons confronté à des question de meilleure maintenance / collaboration / performance.

Vue propose une alternative permettant de mieux gérer la déclaration (**registration**) des composants.

Pour le moment nos composants sont enregistrés Globalement (tous dans main.js), Ils sont disponibles au global dans toute l' app (`createApp()`).

Ce n'est pas forcément idéal quand les composant se complexifient et contiennent beaucoup de code parce que de fait, VUE va charger tous ces composant au lancement de l'application, cela peut poser problème niveau performances.

On peut se retrouver avec une longue liste de composants dans main.js et on n'a pas trop d'indications de l'endroit précis où ils sont utilisés.

Dans notre exemple : le header et user-info sont des composants qui ne sont utilisés qu'une fois.

On va pouvoir les déclarer localement dans les composants dans lesquels on va les utiliser.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component Registration : test avec un composant

Récupérer le projet ([GitHub](#))

Dans main.js :

° Supprimer l'import du composant header **ET** sa déclaration avec la fonction .component()

Dans App.vue :

° Dans script : importer le composant LeHeader

° Dans l'**export default** rajouter une propriété **components** (un objet)

° Dans **components: {}**, à 'le-header' (la chaîne de caractère) on assigne LeHeader

Les Amis : Vue Components Registration

Badges dispo

LOL



admin du site

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component Registration : on déclare tous les composants dans App

Même opération que l'exercice précédent mais pour les composants BadgeList et UserInfo

Les Amis : Vue Components Registration



Badges dispo

NUL PREMIUM toto

LOL

admin du site

[GitHub](#)

Désormais dans notre application, les composants BadgeList, UserInfo et LeHeader sont déclarés **LOCALEMENT** dans le composant App.vue.

Mais on peut également poursuivre la logique en déclarant des composants localement dans n'importe quel composant (selon les niveaux d'imbrication).

C'est plus logique et plus performant, plus maintenable.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Scoped Styles

Le concept de composant repose sur le fait que chaque composant puisse avoir son style CSS, la plupart du temps on a tendance à réaliser un style CSS plutôt global pour toute l'application (ou on utilise un framework CSS) cela peut amplement suffire.

Mais...

Quand on va travailler avec un style Global puis ensuite commencer à faire des styles plus précis au sein des composants, on va potentiellement avoir des problèmes CSS justement.

De base quand une application vue est compilée, par défaut le style d'un composant est rendu Global, il n'affecte pas uniquement le composant lui-même, il peut affecter le reste de l'application

Pour cela VUE adopte un concept de style **scoped**

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : component Style Scoped

- °Dans index.html : mettre en commentaire le CSS de Bootstrap
- °Dans App.vue rajouter dans un <style>, sur les balises html h1 et h2 mettre un font family à impact
- °Voir le résultat dans le navigateur
- °Sur la balise <style> rajouter l'attribut scoped
- °Voir le résultat dans le navigateur
- °Tester dans les autres composants ?

Les Amis : Vue Components style scoped

Badges dispo

NUL

PREMIUM

toto

LOL

PREMIUM

admin du site

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Slots : une autre manière de passer des données dans un composant

Jusqu'ici pour communiquer des données entre les composants nous passons par les props, mais dans certains cas il peut être intéressant de communiquer du contenu Html entre les composants avec les **slots**.

L'idée serait de vouloir transmettre un fragment de template à un composant enfant, et laisser le composant enfant faire le render (le rendu) de ce fragment de template dans son propre template.

La balise <slot> indique l'endroit où le contenu du slot fourni par le parent doit être rendu.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice component slots

Dans le dossier components :

- °Créer un nouveau fichier BaseCard.vue

Dans BaseCard.vue :

- °le template contient une div

- °Le style est **scoped**

- °Copier le style de section qui se trouve dans UserInfo et le placer dans BaseCard, dans BaseCard modifier le sélecteur Css en « div »

- °Modifier le style de la div

- °Dans le template de BaseCard, rajouter dans la div, la balise <slot>

Dans main.js :

- °rajouter l'import du composant BaseCard et le déclarer avec la fonction component() sous le nom 'base-card'

Dans UserInfo.vue :

- °Dans le template, englober à partir du <h3>, le <base-badge> et le <p> dans une balise <base-card>

Dans BadgeList.vue :

- °Mettre la balise <style> en scoped

- °On peut mettre en commentaire le CSS du sélecteur section

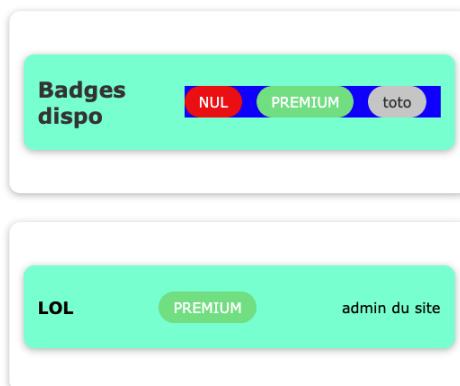
- °Dans le template on va englober le <h2> et la avec notre balise <base-card>

- ° Dans le CSS de la ul rajouter un background-color

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Les Amis : Vue Components slots



Ici on se sert du composant `<base-card>` comme d'un conteneur pour appliquer son style.

[GitHub](#)

Pour le moment on a un seul slot `base-card`, on peut aussi dans certains cas d'utilisations, avoir besoin qu'un composant ai plusieurs slots.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice component multi slots

Dans BaseCard.vue :

- ° Dans le template (au-dessus de <slot>), rajouter une balise <header>, et dans <header>, rajouter une balise <slot> qui possède un attribut name à « header »

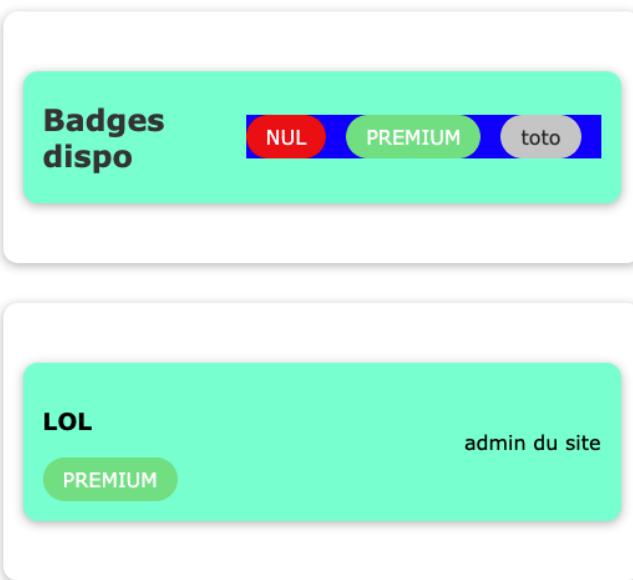
Dans UserInfo.vue :

- ° Dans le template, entourer le h3 et le badge par une balise <template>
- ° Sur cette balise <template> placer une directive v-slot:header
- ° Englober le <p> avec une balise <template> qui a une directive v-slot:default

Dans BadgeList.vue :

- ° Dans le template, englober le <h2> avec une balise <template> avec une directive v-slot:header
- ° englober la avec une balise <template> avec un directive v-slot:default

Les Amis : Vue Components slots



The image shows two examples of Vue components using slots.

Top Example: A card with a green header containing the text "Badges dispo". Below the header is a horizontal bar with three colored circles: red (labeled "NUL"), green (labeled "PREMIUM"), and blue (labeled "toto").

Bottom Example: A card with a green header containing the text "LOL". Below the header is a green button labeled "PREMIUM". In the top right corner of the card, the text "admin du site" is displayed.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

On vient de le voir un composant peut avoir plusieurs slots, autant qu'on veut ... mais avec plusieurs slot il faut permettre à vue de les différencier d'où le fait de nommer les slots (name)

Quand un slot n'a pas de nom Vue va le considérer comme le slot par défaut de ce composant.

```
<template>
  <div>
    <header>
      <slot name="header"></slot>
    </header>
    <slot></slot>
  </div>
</template>
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

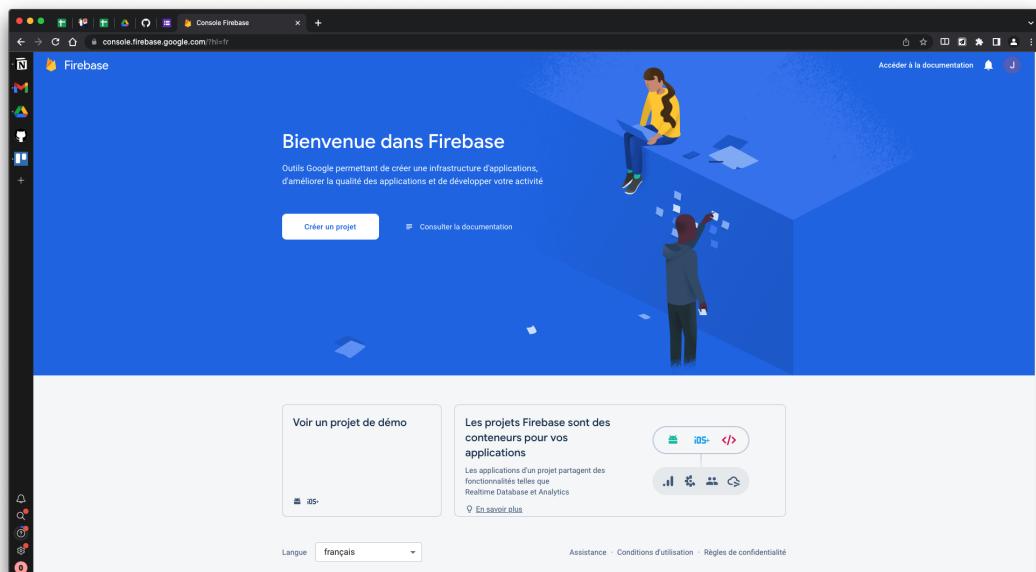
VUE

VUE + Firebase (communiquer avec une BDD)

setup de Firebase (console web)

Rdv à cette adresse <https://console.firebaseio.google.com/>, Se connecter

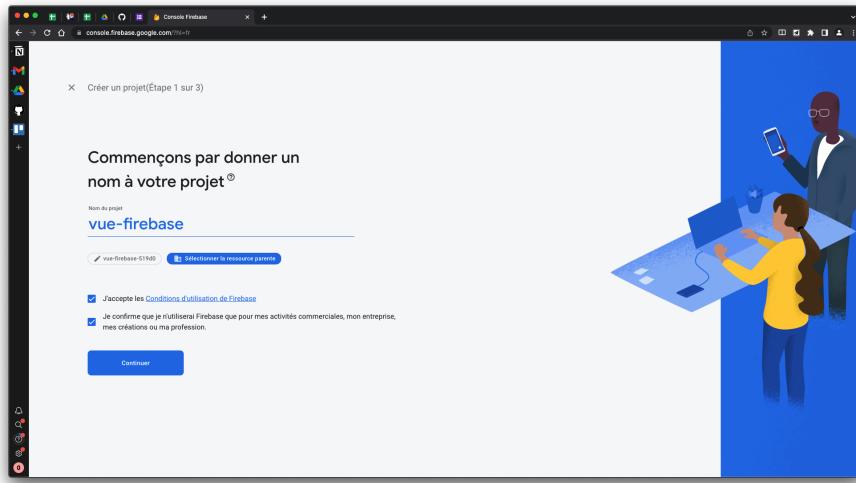
Une fois connecté :



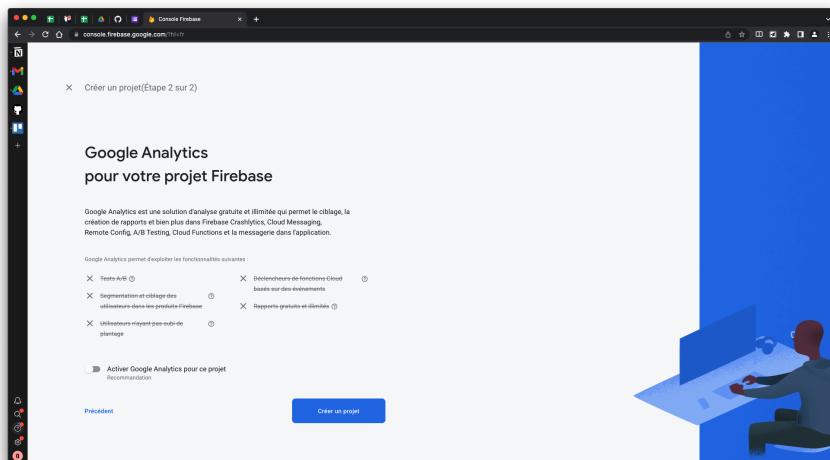
Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

On crée un projet, on donne un nom de projet



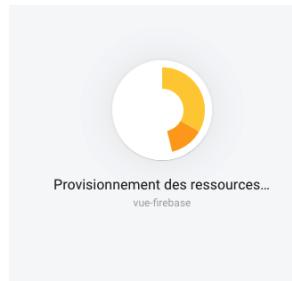
Pour notre app de test pas besoin de google analytics :



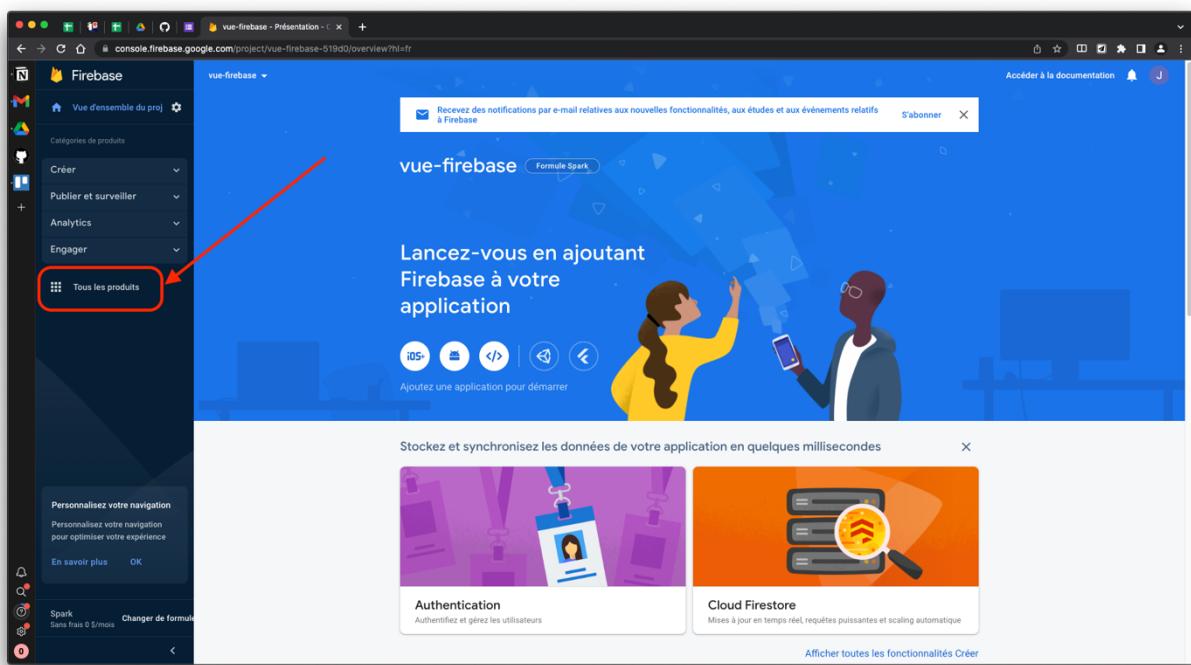
Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Le projet se crée :



On va dans Tous Les Produits pour sélectionner REALTIME DATABASE



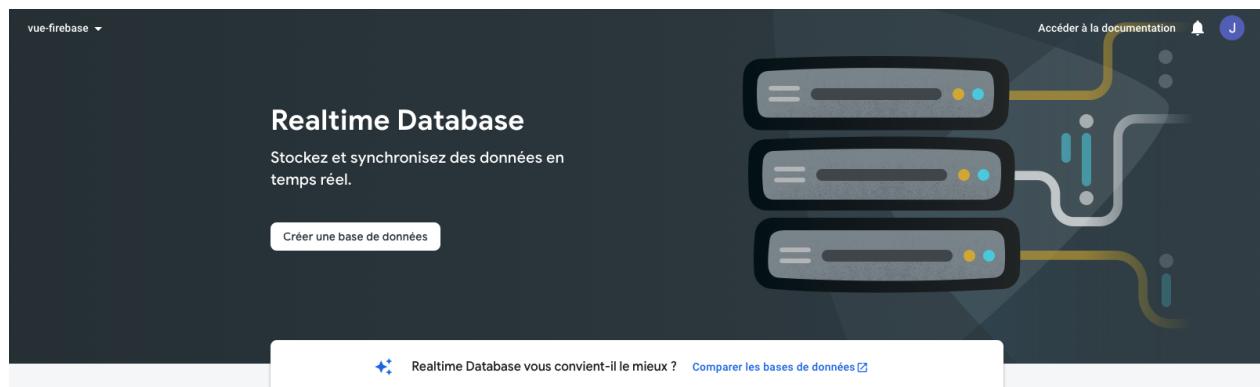
The image shows a screenshot of the Firebase console for a project named "vue-firebase". On the left, there is a sidebar with a red arrow pointing to the "Tous les produits" button under the "Créer" section. The main area shows an overview of the project with sections for "Authentication" and "Cloud Firestore".

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Créer une REALTIME DATABASE



vue-firebase ▾

Realtime Database
Stockez et synchronisez des données en temps réel.

Créer une base de données

Realtime Database vous convient-il le mieux ? [Comparer les bases de données](#)

Emplacement des données (pas important pour notre test)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Configurer une base de données

1 Options de la base de données ————— 2 Règles de sécurité

Le paramètre d'emplacement définit l'endroit où vos données Realtime Database seront stockées.

Emplacement Realtime Database

États-Unis (us-central1)

Annuler Suivant

Mettre la RealTime Database en mode test :

Configurer une base de données

1 Options de la base de données ————— 2 Règles de sécurité

Après avoir défini la structure de vos données, vous devrez spécifier des règles pour les sécuriser.
[En savoir plus](#)

Commencer en mode verrouillé
 Par défaut, vos données sont privées. L'accès client en lecture/écriture ne sera autorisé qu'en fonction de vos règles de sécurité.

Démarrer en mode test
 Par défaut, vos données sont publiques pour permettre une configuration rapide. Toutefois, vous devez modifier vos règles de sécurité dans les 30 jours pour autoriser l'accès client en lecture/écriture sur le long terme.

```
{
  "rules": {
    ".read": "now < 1661378400000", // 2022-8-25
    ".write": "now < 1661378400000", // 2022-8-25
  }
}
```

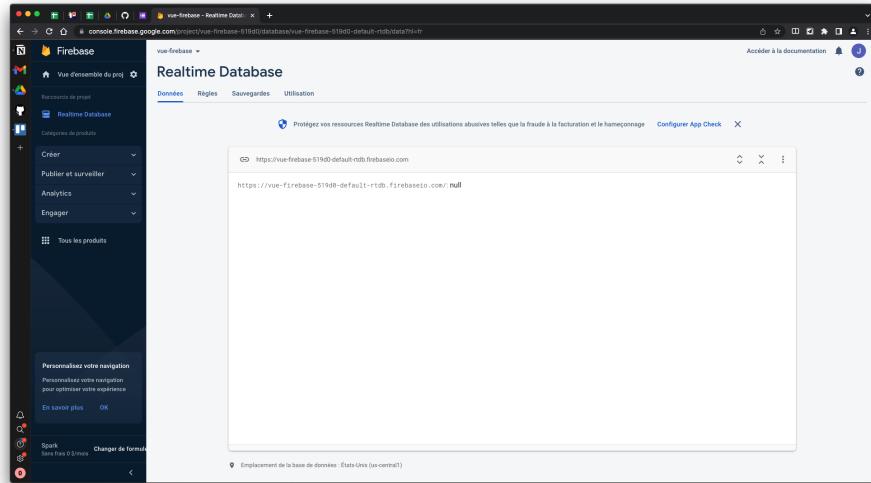
! Par défaut, les règles de sécurité en mode test autorisent tout utilisateur disposant de la référence de votre base de données à afficher, modifier et supprimer toutes les données qu'elle contient pendant les 30 prochains jours

Annuler Activer

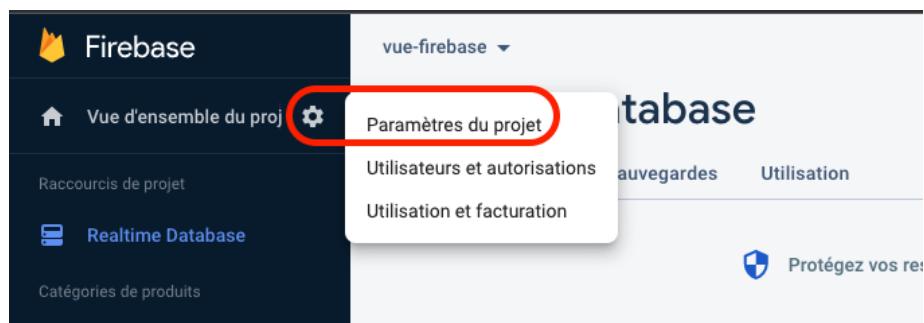
On arrive dans notre RealTime DataBase :

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE



On va connecter notre BDD à notre application web :



En bas de la page Paramètre du projet :

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Vos applications

Votre projet ne comporte aucune application

Pour démarrer, sélectionnez une plate-forme



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Renseignez un nom d'application (pas besoin de firebase Hosting) :

× Ajouter Firebase à votre application Web

1 Enregister l'application

Pseudo de l'application [?](#)

Configurez également **Firebase Hosting** pour cette application. [En savoir plus](#)

Hosting peut également être configuré plus tard, et vous pouvez commencer à l'utiliser sans frais à tout moment.

Enregister l'application

2 Ajouter le SDK Firebase

On arrive sur un écran avec une const firebaseConfig) copier-coller tout le code (mettre de côté), on peut retourner dans la console.

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

SetUp de l'application VUE

Nous allons réaliser une application VUE, qui va gérer des articles (ou users ou produits...), Nous mettrons en place un système de **routing** pour naviguer à travers plusieurs **composants** et nous allons mettre en place un système de **service** pour se **connecter** et **interagir** avec une **BDD**.

Exercice : Setup de l'app

Via le CLI , créer une nouvelle application VUE

vue create vue-firebase

(Default ([Vue 3] babel, eslint).)

Dans le dossier src :

- ° Créer un fichier router.js et un fichier firebase.js
- ° Créer un dossier services

Dans le dossier services :

- ° créer un fichier ArticleDataService.js

Dans le dossier components :

- ° Créer des fichiers pour les composants : AddArticle.vue, Article.vue, ArticlesList.vue

Via la commande npm install dans le terminal, installer Bootstrap (ou équivalent, à voir, si vous voulez).

Dans main.js :

- ° Rajouter l'import de bootstrap

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Router

Nous créons les routes de notre application en passant par ce tableau routes = []

Chaque route sera un objet dans le tableau qui va avoir des propriétés :

- Path : l'URL
- Name (optionnel)
- Component : le composant à charger quand on arrive sur cette URL

La f° createWebHistory permet à vue de créer un historique de navigation au sein de l'application

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : installation / configuration du router

Dans le terminal :

- ° Installer le router de vue dans le projet (npm install vue-router@4)

Dans main.js :

° Rajouter l'import du router import **router** from './router'

° Utiliser la fonction use en lui passant router

```
createApp(App).use(router).mount('#app')
```

Dans router.js :

° Importer les f° createWebHistory et createRouter from "vue-router"

```
import { createWebHistory, createRouter } from "vue-router";
```

° Créer une const **routes** à laquelle on assigne un **tableau**

° Dans ce tableau rajouter un objet JS {} dans lequel on a une propriété

path à laquelle on assigne "/",

alias à laquelle on assigne "/articles",

name à laquelle on assigne "articles",

component à laquelle on assigne : () => import("./components/ArticlesList.vue")

° Dans ce tableau **routes** rajouter un 2nd objet JS {} dans lequel on a une propriété

path à laquelle on assigne "/add",

name à laquelle on assigne "add",

component à laquelle on assigne : () => import("./components/AddArticle")

° Après le tableau des **routes** : créer une const router à laquelle on assigne la f° createRouter()

° la f° createRouter(), prend en paramètre un objet JS {}

° Dans cet objet JS, on a :

une propriété history à laquelle on assigne la f° createWebHistory()

une propriété routes

À la fin on fait un export default de router

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

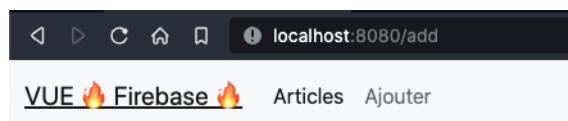
VUE

Exercice : Utilisation du router (ajout navbar)

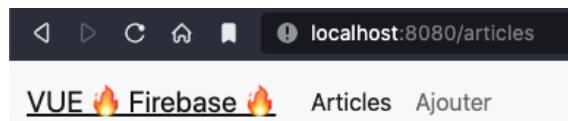
Dans App.vue :

- ° Garder juste la balise <template> et la <div id='app'>
- ° Dans la <div class='app'>, rajouter une navbar avec 3 liens (home , articles, ajouter)
- ° Entourer un lien <a> avec une balise <router-link></router-link>
- Sur <router-link> rajouter un attribut **to="/"**
- ° Sur le 2è lien <a> remplacer la balise <a> par <router-link></router-link>
- Sur <router-link> rajouter un attribut **to="/articles"**
- ° Sur le 3è lien <a> remplacer la balise <a> par <router-link></router-link>
- Sur <router-link> rajouter un attribut **to="/add"**
- ° En dessous de la <nav> rajouter une balise <router-view />

(aller mettre du template dans les composants ArticlesList et AddArticle)



Ajouter Articles



Articles LIST

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

[GitHub](#)

Installation / configuration de firebase

Techniquement notre BackEnd C'est Firebase, là-dedans se trouve notre Base de données (RealTime Database), que l'on installe Firebase ou non, on va dans tous les cas contacter une API de manière asynchrone.

En JS il y a plusieurs façons de faire des fonctions asynchrones principalement **async await** et **then().catch()**

Quand on installe Firebase dans notre projet via la commande npm (ou autre), on installe juste la bibliothèque JS de firebase, ce qui donne accès aux fonctions de firebase.

Pour interagir avec le BDD nous utiliserons des fonctions en mode asynchrone comme ceci :

```
deleteArticle() {
    ArticleDataService.delete().then(Du code).catch(Du code);
},
```

Installer Firebase dans le projet

npm install firebase@8.10.0

(Si jamais la V9 pose problème)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice Config de la Connexion à Firebase

Dans firebase.js :

- ° Rajouter les import de firebase :

```
import firebase from 'firebase/app';
```

```
import "firebase/database";
```

- ° Rajouter la const firebaseConfig (la variable de config dispo dans le site firebase)

- ° Sur firebase, utiliser la f° initializeApp() qui prend en paramètre firebaseConfig

- ° Enfin on fait un export default de firebase, sur laquelle on utilise la f° database()

Dans ArticleDataService.js :

- ° import firebase from './firebase';

- ° Créer une const db à laquelle on assigne firebase sur laquelle on utilise la f° .ref().

- ° ref() prend en paramètre "/articles"

- ° Créer une classe ArticleDataService {} (vide pour le moment)

- ° Ensuite on fait un export default d'une nouvelle instance de cette classe ArticleDataService

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : Mise en place du Service (utilisation de class JS) via ajout en BDD

Dans ArticleDataService.js (dans la CLASS):

- ° Créer une f° getAll() dans laquelle on retourne db
- ° Créer une f° create() qui prend article en paramètre et dans laquelle on retourne db sur laquelle on utilise la f° push(article)
- ° Créer une f° update() qui prend key et value en paramètre et dans laquelle on retourne db sur laquelle on utilise la f° child(key) et la f° update(value)
- ° Créer une f° delete() qui prend key en paramètre et dans laquelle on retourne db sur laquelle on utilise la f° child(key) et la f° remove()
- °Créer une f° deleteAll() dans laquelle on retourne db sur laquelle on utilise la f° remove()

Dans AddArticle.vue dans la partie <template> :

- ° mettre en place un formulaire avec 2 inputs (1 pour le **title**, 1 pour la **description**) et un bouton

Dans AddArticle.vue dans la partie <script> :

° Importer ArticleDataService

Dans l'export default :

- ° Une propriété **name** à laquelle on assigne « add-article »

° Dans **data(){}**, on **retourne** dans un objet : une variable **submitted** initialisée à false et un objet

article {}

- ° Dans l'objet **article** on a des propriétés :

title et **description** initialisé avec “ (string vide quoi)

published initialisée à false

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

°Dans **methods**:

° Rajouter une f° saveArticle () dans laquelle on déclare une variable **data** qui est un objet (vide pour le moment)

° On va remplir cet objet **data** :

On assigne à **title** le this.article.title

On assigne à **description** le this.article.description

On assigne à **published**, false

°Après cet objet , toujours dans la f° saveArticle:

°Sur **ArticleDataService** utiliser la f° create() qui prend data en paramètre **data**

chainé à create() utiliser la f° .then()

chainé à then() utiliser la f° .catch() ca nous donne cà pour le moment

ArticleDataService.create(data).then().catch();

°Dans les paramètres de **then()** faire une fonction fléchée anonyme

()=>{}

dans cette fonction fléchée on va :

assigner true à **submitted** (c'est dans les data de vue)

faire un console log qui dit 'article crée'

°Dans les paramètres de **catch()**, faire une fonction fléchée anonyme avec une variable error en paramètre

error=>{}

Dans cette dernière fonction fléchée: on fait un console log de **error**

Toujours dans methods (toujours dans AddArticle.vue) :

°Rajouter une methods newArticle () dans laquelle :

° on passe la variable submitted à false

° On assigne à article un objet dans lequel :

° On assigne à title :“ “

° On assigne à description :“ “

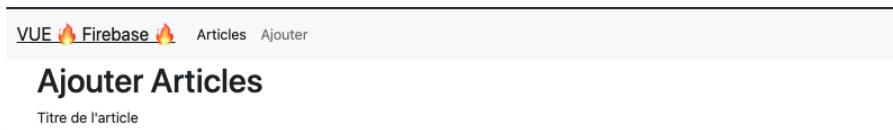
° On assigne à published : false

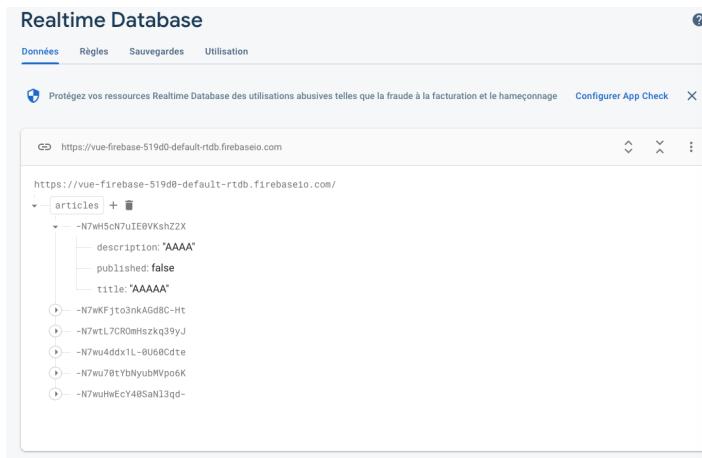
Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans AddArticle.vue dans la partie <template> :

- ° Sur le bouton ajouter placer un v-on qui réagit au click et fait référence à la f° saveArticle
- ° sur le un input place un v-model pour le lier avec article.title
- ° sur le un input place un v-model pour le lier avec article.description





[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Cycle de vie des composants (LifeCycle Hooks)

Dans l'exercice suivant nous allons déclencher des fonctions À un certain moment du cycle de vie d'un composant :

En arrière fond VUE gère avec précision la vie d'un composant en effet lorsque nous arrivons sur une page qui appelle un composant celui-ci passe par plusieurs états et VUE met à disposition des fonction pour exécuter n'importe quel code JS à certains moment de la vie du composant :

Si vous vous souvenez dans chaque application VUE, dans main.js on utilise une fonction **mount()**, pour monter le composant sur une div du template (pour que vue prenne le contrôle de cette <div>)
Dans l'exercice ci-après nous allons utiliser la fonction **mount()** pour signifier à notre composant (**AU MOMENT** où celui-ci est **MONTÉ** dans l'application) de surveiller la BDD (avec la fonction **.on()** pour firebase).

Ensuite nous allons utiliser la fonction **beforeUnmount()** pour signifier à notre composant (Juste **AVANT LE MOMENT** où celui-ci est **DÉMONTÉ** (donc juste avant qu'il ne soit plus afficher dans la page)) de ne plus surveiller notre BDD avec la fonction **.off()** de firebase

Sur la page suivante vous trouverez un schéma récapitulatif des états du cycle de vie d'un composant
Les fonction de VUE sont assez explicites :

beforeCreate : avant que le composant ne soit créé

created : Quand le composant est créé

beforeMount : Avant qu'il soit monté (placé) dans l'application

mounted : Quand il vient d'être monté (placé) dans l'application

beforeUpdate : Avant que le composant se mette à jour

update : Quand le composant se met à jour

beforeUnmount : Avant que le composant ne soit (démonté / retiré) de l'application

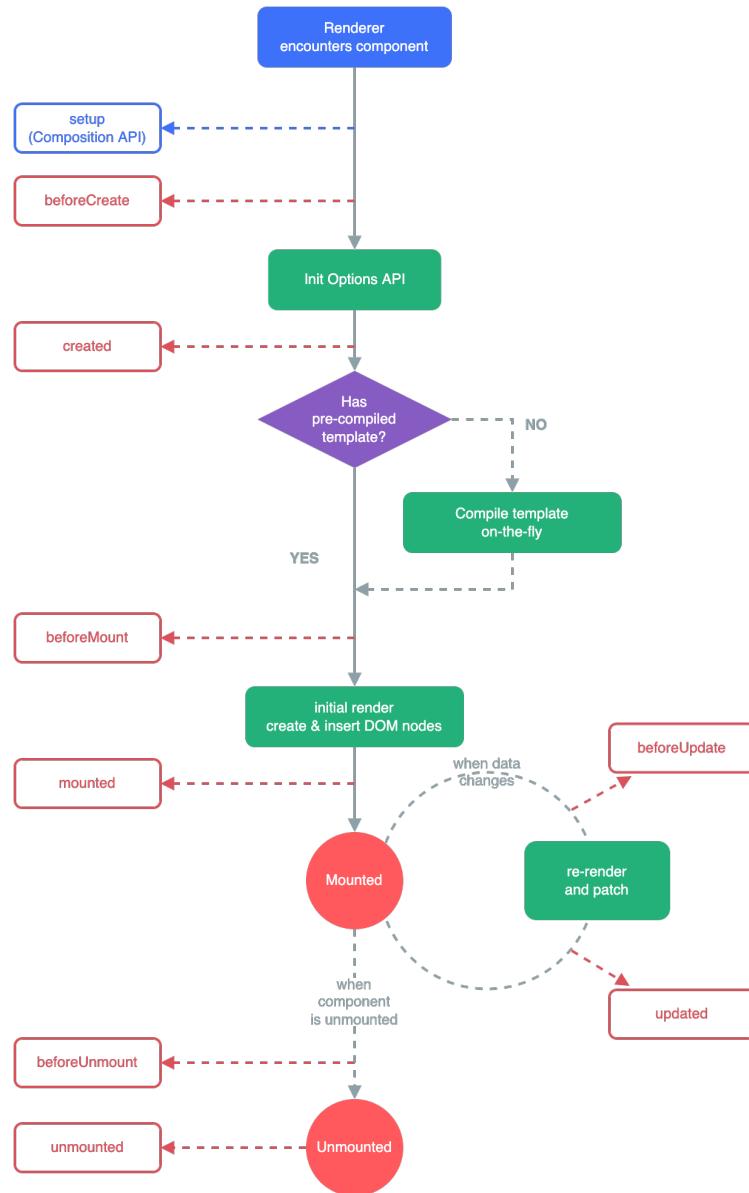
unmount : Quand le composant est démonté (retiré) de l'application

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice : Liste des articles

Dans ArticlesList.vue <script> :

° Importer ArticleDataService

° Importer ArticleDetails from ./Article.vue (on peut importer un composant sous un autre nom)

Dans l'export default :

° Une propriété name : "articles-list"

° Une propriété component : { ArticleDetails }

° Dans data() {} :

articles, un tableau vide

currentArticle à null

currentIndex à -1

Dans ArticlesList.vue <script>(methods)

° Créer une methods onDataChange(), qui prend en paramètre items

° Dans cette f° :

Créer une variable _articles, un tableau vide

(le _ devant c'est pour la différencier de articles que l'on a dans data)

Sur items, utiliser la f° forEach(), qui prend en param une f°=> qui elle a en param item

Dans la f°=> :

° Créer une variable key, on lui assigne item.key

° Créer une variable data, on lui assigne item.val()

° Sur _articles, utiliser la f° push() qui contient un objet

° Dans cet objet : à

key on lui assigne key,

À title on assigne data.title

À description on assigne data.description

À published on assigne data.published

° En dehors du forEach (mais tjs dans onDataChange), à this.articles, on assigne _articles

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans ArticlesList.vue <script>(methods) :

- °Créer une methods **refreshList()**,dans laquelle :
 - ° On assigne à currentArticle, null
 - ° On assigne à currentIndex, -1

- °Créer une methods **setActiveArticle()**,qui prend en param **article** et **index**, Dans cette f°:
 - °On assigne à currentArticle, article
 - °On assigne à currentIndex, index

Dans ArticlesList.vue <script>(methods)

- °Créer une methods **removeAllArticles()**,dans laquelle :
 - °Sur ArticleDataService on utilise la f° deleteAll()
 - °Chainé à deleteAll(), utiliser la f° then() puis catch()
 - on a donc ceci : ArticleDataService.deleteAll().then().catch()
 - °Dans les paramètres de then() on a une fonction fléchée
 - ° Dans cette f° =>, on execute la f° refreshList
 - °Dans les paramètres de catch() on a une fonction fléchée qui a en paramètre e
 - °Dans cette f°=>, on console.log(e)

Dans ArticlesList.vue <script>(en. Dehors des methods) :

- °Créer une f° **mounted()**,dans laquelle :
 - °Sur ArticleDataService on utilise la f° getAll()
 - ° Chainé à getAll(), utiliser la f° on() comme ceci : ArticleDataService.getAll().on();
 - ° la fonction on() prend en 1er param “value” et en 2nd param, this.onDataChange
- °Créer une f° **beforeUnmount()**,dans laquelle :
 - °Sur ArticleDataService on utilise la f° getAll()
 - ° Chainé à getAll(), utiliser la f° off() comme ceci : ArticleDataService.getAll().off();
 - ° la fonction off() prend en 1er param “value” et en 2nd param, this.onDataChange

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

ArticlesList.vue <template> :

```
<template>
<div class="container list row">
  <h1>Articles LIST</h1>
  <div class="col-md-6">
    <ul class="list-group">
      <li></li>
    </ul>
    <button class="m-3 btn btn-sm btn-danger">Tout Suppr.</button>
  </div>
  <div class="col-md-6">
    <div>
      <article-details></article-details>
    </div>
    <div>
      <p>Veuillez selectionner un article.</p>
    </div>
  </div>
</div>
</template>
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans **ArticlesList.vue <template>** :

°Sur la **** :

- ° un v-for qui parcourt le tableau **articles** avec une variable **unArticle** et l'index
- ° un v-bind de key pour le lier à l'index du v-for
- ° un v-on qui réagit au click et exécute la methods **setActiveArticle** avec en paramètre **unArticle** et index

°Dans la **** : avec l'interpolation afficher le title de **unArticle**

°Sur le **<button>** Tout Suppr : réagir au click pour appeler la methods **removeAllArticles**

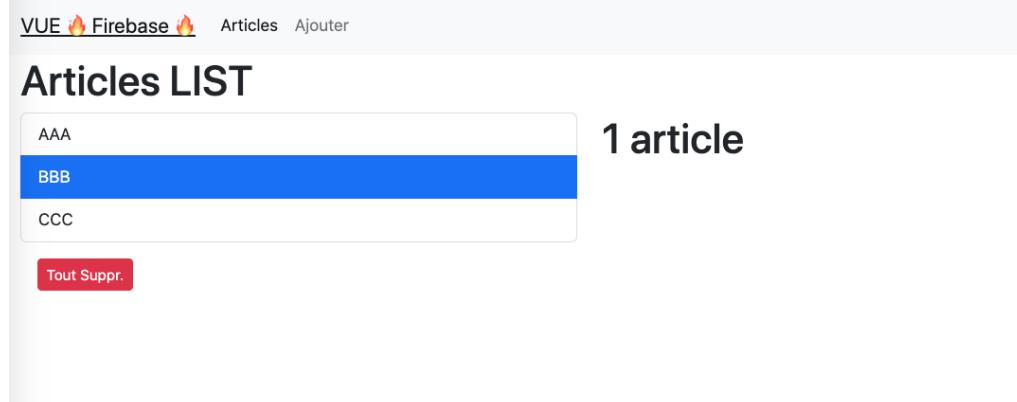
°Sur la **<div>** (juste au-dessus de **<article-details>**) : un v-if qui vérifie si **currentArticle** est définie.

°Sur la **<div>** (juste au-dessus u **<p>**) : un v-else

°Sur la balise **<article-details>** :

°un v-bind de article pour le lier à **currentArticle**

°(pour plus tard)réagir à l'évènement **refreshList** pour appeler la **methods refreshList**



The screenshot shows a user interface for managing articles. At the top, there's a navigation bar with links for 'VUE 🔥 Firebase 🔥', 'Articles', and 'Ajouter'. Below it, the title 'Articles LIST' is displayed. The main area contains a list of items: 'AAA' (in a grey box), 'BBB' (highlighted in blue), and 'CCC' (in a grey box). To the right of the list, the text '1 article' is shown. At the bottom left, there's a red button labeled 'Tout Suppr.'

[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Exercice :Création du composant article

Dans Article.vue <script> :

° Importer ArticleDataService

° Dans l'export default :

Une propriété name : "article" ou « un-article » (à voir si on a des warning en console)

Une propriété props : ["article"]

Dans data() { } :

° currentArticle à null

° message à “ ”

Dans watch { } :

article à laquelle on assigne une f° qui prend en param article, dans cette f°:

On assigne à this.currentArticle, { ...article }

On assigne this.message, “ ”

Dans methods: { } :

°Créer une methods **updatePublished()** qui prend en param status,

°Dans cette f° :

°Sur ArticleDataService utiliser la f° update() qui prend

°en 1er param this.currentArticle.key

°en second param un objet {} dans lequel

°à published, on assigne status

°Chainé à update(), utiliser la f° then() qui prend en param une fonction fléchée, dans la fonction :

°À this.currentArticle.published assigner status

°À this.message assigner « Le status à bien été MAJ »

°Chainé à then(), utiliser la f° catch() qui prend en param une fonction fléchée qui a en param e

° on console.log(e)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans methods: { } :

- °Créer une methods **updateArticle()** ,Dans cette f° :
- °Créer une const data, un objet, dans lequel :
 - °À title on assigne this.currentArticle.title
 - °À description on assigne this.currentArticle.description
- Après l'objet data :
- °Sur ArticleDataService utiliser la f° update() qui prend
 - °en 1er param this.currentArticle.key
 - °en second param data
 - °à published, on assigne status
- °Chainé à update(), utiliser la f° then() qui prend en param une fonction fléchée, dans la fonction
 - °À this.message assigner « L'article a bien été MAJ »
- °Chainé à then(), utiliser la f° catch() qui prend en param une fonction fléchée qui a en param e
 - ° on console.log(e)

Dans methods: { }

- °Créer une methods **deleteArticle()** ,Dans cette fonction :
- °Sur ArticleDataService utiliser la fonction delete() qui prend en param this.currentArticle.key
- °Chainé à delete(), utiliser la f° then() qui prend en param une fonction fléchée dans laquelle :
 - °Avec la f° \$emit, on EMET l'évent "refreshList"
- °Chainé à then(), utiliser la f° catch() qui prend en param une fonction fléchée qui a en param e
 - ° on console.log(e)

En dehors de methods{} :

- °Créer une f° **mounted()** dans laquelle,
 - °À this.message on assigne “ ”
 - °À this.currentArticle on assigne { ...this.article }

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans Article.vue <template> :

```

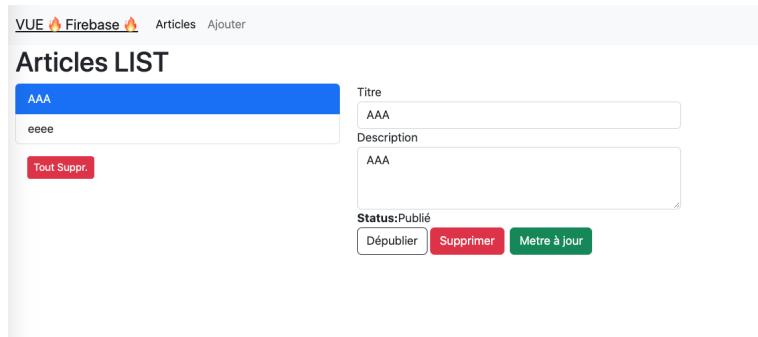
•   <template>
•     <div class="edit-form">
•       <form>
•         <div class="form-group">
•           <label for="title">Titre</label>
•           <input type="text" class="form-control" id="title"/>
•         </div>
•         <div class="form-group">
•           <label for="description">Description</label>
•           <textarea class="form-control" rows="3"></textarea>
•         </div>
•
•         <div class="form-group">
•           <label><strong>Status:</strong></label>
•         </div>
•       </form>
•
•       <button class="btn btn-outline-dark">Dépublier</button>
•       <button class="btn btn-info mx-1">Publier</button>
•       <button class="btn btn-danger mx-1">Supprimer</button>
•       <button type="submit" class="btn btn-success mx-1">Mettre à
jour</button>
•       <p>{{ message }}</p>
•     </div>
•
•     <div>
•       <p>Cliquez sur un article Por Favor</p>
•     </div>
•   </template>
```

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Dans Article.vue <template> :

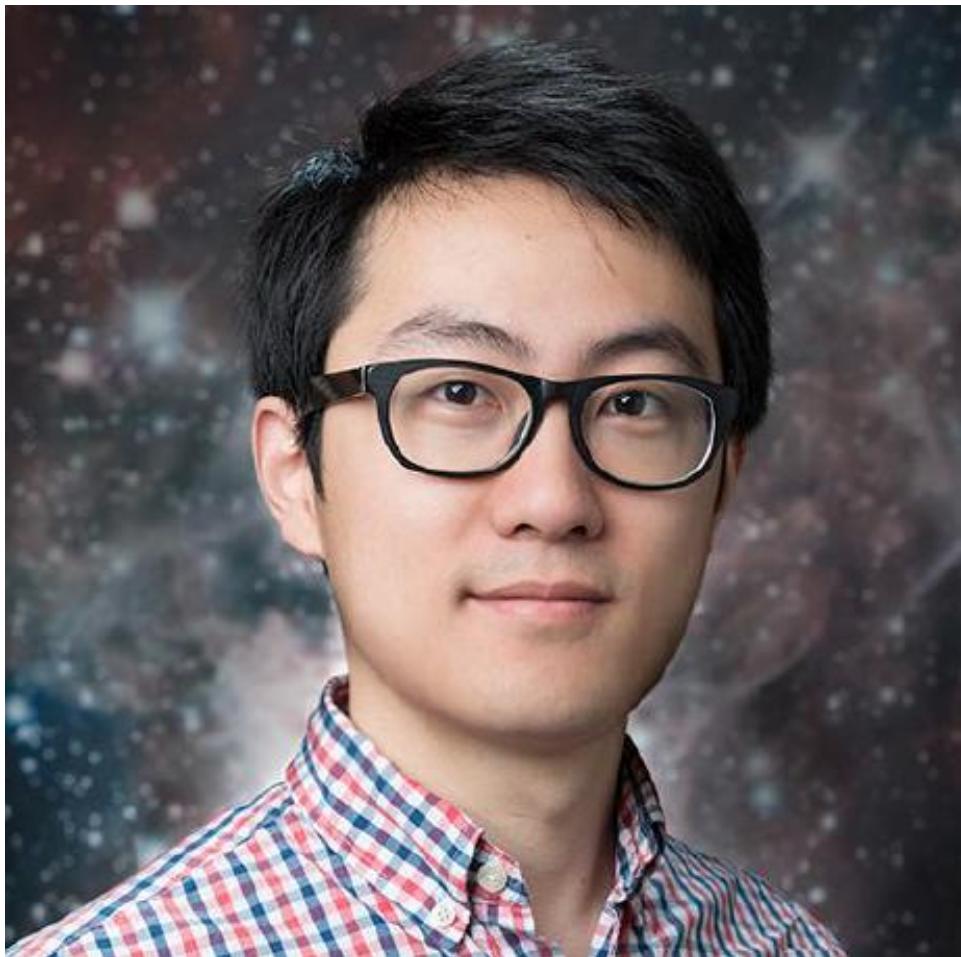
- ° Sur la 1^{ère} <div> un v-if vérifie si currentArticle est définie
- ° Sur l'input de title, placer un v-model relier à currentArticle.title
- ° Sur textarea de description, placer un v-model relier à currentArticle.description
- ° Dans le <label> avec une condition ternaire sur currentArticle.published on affiche 'publié' ou 'en attente'
- ° sur le bouton Dépublier
 - ° un v-if vérifie si currentArticle.published est définie (true ou false)
 - ° on réagit au click pour exécuter la f° updatePublished en passant false en paramètre
- ° sur le bouton Publier
 - ° un v-else
 - ° on réagit au click pour exécuter la f° updatePublished en passant true en paramètre
- ° sur le bouton Supprimer
 - ° on réagit au click pour appeler la f° deleteArticle
- ° sur le bouton Mettre à jour
 - ° on réagit au click pour appeler la f° updateArticle
- ° Dans le <p> on affiche la data message
- ° Sur la dernière <div> placer un v-else (rappel 1^{ère} div)



[GitHub](#)

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.
	Marc CECCALDI Coordinateur Filière Développement		

VUE



Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		

VUE

Auteur(s)	Relu, validé et visé par :	Date de création :	Date dernière MAJ :
<u>Mathieu Mithridate</u>	Jérôme CHRETIENNE : Resp. Secteur Tertiaire & Numérique	25/01/2021	
	Florence CALMETTES : Coordinatrice Filière Syst. & Réseaux		<i>Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.</i>
	Marc CECCALDI Coordinateur Filière Développement		