

Vue JS

Le Router

Vue Router est le router officiel pour Vuejs. Il s'intègre aisément avec Vue.js pour faire des applications mono page avec Vuejs. On retrouve principalement :

- Vues et routes imbriquées
- Modulaire, configuration basée sur les composants
- Paramètres de route, de requête
- Gestion fine de la navigation
- ...

Vous devez d'abord installer le module router

```
npm install vue-router
```

Nous allons ensuite créer un fichier index.js dans un dossier router. Et y rajouter du code

```
// src : router/index.js
import Vue from 'vue'
import Router from 'vue-router'
import Home from '@components/home.vue'
import Foo from '@components/foo.vue'
import Bar from '@components/bar.vue'
```

```
Vue.use(Router))
```

Puis y ajouter aussi

```
// src : router/index.js
export default new Router({
  routes: [
    { path: '/', name: 'Home', component: Home },
    { path: '/foo', name: 'Foo', component: Foo },
    { path: '/bar', name: 'Bar', component: Bar }
  ]
})
```

Enfin il faut charger cette configuration dans le fichier main.js du projet:

```
// src : main.js
import router from './router'
```

Puis modifier l'instance pour ajouter le router :

```
// src : main.js
new Vue({
  router,
```

```
render: h => h(App),
}).$mount('#app')
```

Enfin pour tester rapidement le router, créer les composants de l'exemple Foo et Bar. Et générez deux liens afin de pouvoir vous déplacer de page en page.

Dans le fichier App.vue, mettre dans le HTML les lignes:

```
<router-link to="/foo">Aller à Foo</router-link><br/>
<router-link to="/bar">Aller à Bar</router-link>
<router-view></router-view>
```

- Les deux lignes `router-link` permettent de générer automatiquement les liens cliquables. Le lien `router-view` est le système de relation de l'affichage d'un composant par rapport à une route.
- `router-view` est remplacé par le composant relatif au routing défini dans le fichier de routing

Exercice: 20min

- Mettez en place un router simple avec deux pages et deux composants
- Faites en sorte de pouvoir naviguer entre les pages

Route paramétrée

Dans de nombreux cas d'utilisation, vous avez besoin de passer une donnée d'une page à une autre

Pour ce faire, on utilise habituellement les routes paramétrées :

- `www.facebook.com/user/3`
- `www.facebook.com/user/3000`

Ici le dernier élément est dynamique. On représente ces paramètres dans notre routing avec `:paramName`.

Par exemple, pour la route bar `/bar/:id`

Une fois la route modifiée, il suffit de lire la données passée en paramètre. C'est là que les cycles de vie rentrent en jeu.

- Ou au moins le cycle `created`
- Depuis ce cycle de vie, vous pouvez lire le contenu du paramètre `id` en écrivant

```
created() {
  console.log(this.$route.params.id);
}
```

Exercice: 30min

- Créez différentes routes
- Gérez des paramètres
- Lisez et traiter ces paramètres
- Faites une liste de choix avec des comptes utilisateurs github. En fonction du pseudo sélectionné, la page affichera le profil de la personne.
`https://api.github.com/users/:pseudo`

****** Exemple : `exercices/vuejs-routes`
