

# Comment intégrer React JS à Moodle

*Cet article a été écrit par l'équipe du Service national du RÉCIT de la formation à distance*

*Septembre 2022*

## Table des matières

Introduction .....	2
Pourquoi intégrer React JS à Moodle .....	2
Installation de logiciels.....	3
Intégration de React JS à Moodle .....	3
Web API .....	4
Des exemples de plugiciels.....	5
Générer le zip d'installation .....	6
Des résultats obtenus .....	7
RÉCIT Autolink .....	7
Éditeur HTML Bootstrap RÉCIT .....	7
Apprentimètre RÉCIT .....	8
Cahier de traces .....	8
Conclusion .....	9
Références .....	9

# Introduction

Le développement d'interfaces d'utilisateurs est un grand enjeu dans le domaine des logiciels. La facilité d'utilisation, son efficacité à accomplir une action déterminée, le coût pour son développement, sa pérennité, etc., sont des facteurs qui peuvent avoir un impact direct sur l'efficacité d'un logiciel. Ceci est encore plus marqué en éducation, principalement lorsque le public cible est jeune. En conséquence, le développement d'interfaces pour les environnements numériques d'apprentissages tels que [Moodle](#) doit être bien réfléchi et organisé.

La technologie choisie pour développer un logiciel est un autre facteur important. Sans aucun *framework* (bibliothèque) standardisant le développement, la conception de logiciels volumineux et complexes avec les langages HTML, CSS et JavaScript devient très ardu. Le choix d'une bibliothèque répondant aux besoins précis est à considérer. Le Service national du RÉCIT de la formation à distance a choisi la bibliothèque [React JS](#) pour ses développements dans Moodle.

## Pourquoi intégrer React JS à Moodle

Moodle est peut-être l'ENA (environnement numérique d'apprentissage) gratuit le plus connu et utilisé au monde. Il s'agit d'un système robuste et flexible présentant un avantage remarquable : sa capacité d'être améliorée grâce à l'intégration de plugiciels.

React JS est une bibliothèque JavaScript efficace et flexible pour concevoir des interfaces utilisateurs. Développée par la compagnie Meta, elle est utilisée par de grands joueurs tels que Facebook, Netflix, Uber, entre autres.

React JS utilise des composants qui permettent d'abstraire la syntaxe de code HTML CSS et JavaScript. Le fichier React JS obtenu est compilé en fichiers *.js* et *.css* et intégrés dans une page web.

Voici quelques avantages d'utiliser React JS dans le développement de Moodle :

1. Augmentation de la vitesse d'échanges de données : le minimum d'information est échangé entre l'application et le serveur.
2. Interfaces intuitives et attrayantes : l'utilisation de JSX (l'extension du langage JavaScript pour React JS) facilite l'intégration d'autres bibliothèques ou composants, tels que Bootstrap, Font Awesome, etc.
3. Pérennité de l'application : le code source, bien encapsulé dans des composants, est plus facile à lire et à entretenir.
4. Augmentation de la vitesse d'implémentation de plugiciels (réduction de coûts de projets avec la réduction du nombre d'heures de programmation).
5. Réutilisation du code source facilitée.

# Installation de logiciels

Voici les logiciels de base à être installés pour commencer le développement. Présentement, nous utilisons les versions suivantes :

- [Node JS](#) – v14.17.4
- React JS – v17.0.2
- [Parcel JS](#) – v1.12.5
- Moodle – v3.9.16

Vous devez tout d'abord installer Node JS/npm sur votre poste.

## Intégration de React JS à Moodle

Tout d'abord, il faut créer un *div* vide dans le plugiciel où l'on veut placer l'application React JS et lui donner un id composé du nom de l'application. Celui-ci permettra d'intégrer l'application React, dans une page Moodle. Par exemple :

```
<div id="pluginname_react"></div>
```

Pour intégrer React JS à Moodle, un module spécial appelé **Parcel** est utilisé. Il faut donc installer React JS et Parcel :

- Dans le répertoire du plugiciel, ouvrir un terminal et taper les lignes de commandes suivantes :

```
npm install -g create-react-app  
npx create-react-app react_app  
cd react_app  
npm install parcel-bundler@1.12.5
```

Cette opération créera un dossier dans votre plugiciel *react\_app*.

- Modifier le fichier *package.json* pour utiliser le module Parcel comme ceci:

```
"scripts": {  
  "test": "echo \"Error: no test specified\" && exit 1",  
  "start": "parcel ./src/index.js --out-dir ./build",  
  "build": "parcel build ./src/index.js --out-dir ./build"  
},
```

- Repérer le code ci-dessous dans le fichier index.js :

```
const root = ReactDOM.createRoot(document.getElementById('root'));
```

- Remplacer 'root' par 'pluginname\_react'.

```
const root = ReactDOM.createRoot(document.getElementById('pluginname_react'));
```

Cette opération permet que l'application React soit rendue dans le div créé précédemment.

Pour charger les fichiers de React dans Moodle :

- Ajouter ce bout de code juste avant d'afficher notre div vide :

```
$PAGE->requires->css(new moodle_url($CFG->wwwroot .  
'/local/pluginname/react_app/build/index.css'), true);  
$PAGE->requires->js(new moodle_url($CFG->wwwroot .  
'/local/pluginname/react_app/build/index.js'), true);
```

- Dans le terminal, taper *npm start* :

```
npm start
```

Et voilà ! Vous pouvez maintenant commencer à développer votre application React !

- Taper *npm built* :

```
npm built
```

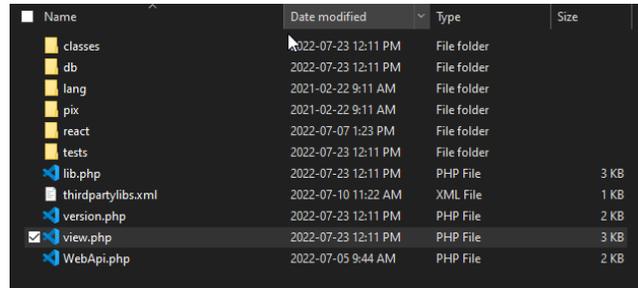
React JS va générer un fichier *.js* et un fichier *.css* dans le dossier *build* qui sera chargé par Moodle.

## Web API

Pour que l'application React JS puisse interagir avec Moodle, il est nécessaire de créer un Web API. L'application React JS pourra donc appeler ce service pour interagir avec le serveur. Moodle a une page d'explication pour créer un API disponible [ici](#).

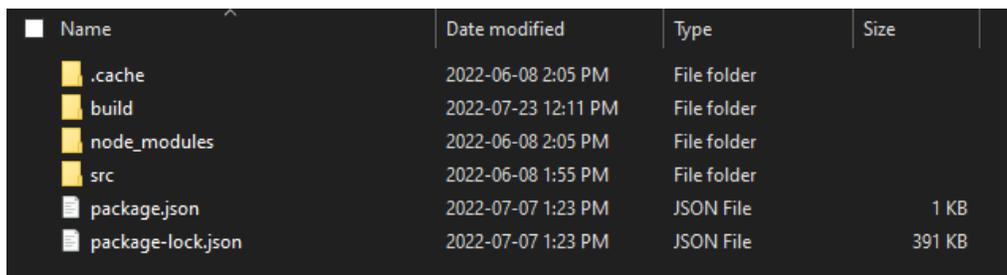
# Des exemples de plugiciels

Des plugiciels de type *local* et *module* :



Name	Date modified	Type	Size
classes	2022-07-23 12:11 PM	File folder	
db	2022-07-23 12:11 PM	File folder	
lang	2021-02-22 9:11 AM	File folder	
pix	2021-02-22 9:11 AM	File folder	
react	2022-07-07 1:23 PM	File folder	
tests	2022-07-23 12:11 PM	File folder	
lib.php	2022-07-23 12:11 PM	PHP File	3 KB
thirdpartylibs.xml	2022-07-10 11:22 AM	XML File	1 KB
version.php	2022-07-23 12:11 PM	PHP File	2 KB
view.php	2022-07-23 12:11 PM	PHP File	3 KB
WebApi.php	2022-07-05 9:44 AM	PHP File	2 KB

Figure 1 – Le fichier *view.php* qui affichera le *div vide*.



Name	Date modified	Type	Size
.cache	2022-06-08 2:05 PM	File folder	
build	2022-07-23 12:11 PM	File folder	
node_modules	2022-06-08 2:05 PM	File folder	
src	2022-06-08 1:55 PM	File folder	
package.json	2022-07-07 1:23 PM	JSON File	1 KB
package-lock.json	2022-07-07 1:23 PM	JSON File	391 KB

Figure 2 – Le dossier de l'application *React*.

```
require_login();

// Globals.
$PAGE->set_url("/local/recitdashboard/view.php");
$PAGE->requires->css(new moodle_url($CFG->wwwroot . '/local/recitdashboard/react/build/index.css'), true);
$PAGE->requires->js(new moodle_url($CFG->wwwroot . '/local/recitdashboard/react/build/index.js'), true);
local_recitdashboard_strings_for_js();

// Set page context.
$PAGE->set_context(\context_system::instance());

// Set page layout.
$PAGE->set_pagelayout('mydashboard');

$PAGE->set_title(get_string('pluginname', 'local_recitdashboard'));
$PAGE->set_heading(get_string('pluginname', 'local_recitdashboard'));

echo $OUTPUT->header();
$courseId = (isset($_GET['courseId']) ? $_GET['courseId'] : 0);
$recitDashboard = new MainView($USER, $courseId);
$recitDashboard->display();

echo $OUTPUT->footer();
```

Figure 3 – L'intégration des fichiers *.js* et *.css* dans le *view.php* de Moodle.

## Des plugiciels de type Atto :

```
Y.namespace('M.atto_reciteditor').Button = Y.Base.create('button', Y.M.editor_atto.EditorPlugin, [], {
  /**
   * Add the buttons to the toolbar
   *
   * @method initializer
   */
  initializer: function() {
    this.addButton({
      title: 'htmleditor',
      icon: 'html',
      iconComponent: 'atto_reciteditor',
      callback: this.openHtmlEditor,
      buttonName: 'htmleditor'
    });
  },

  globalVars: {popup: null},

  openHtmlEditor: function(e) {
    e.preventDefault();

    // if the reference exists and the window is not closed so we can bring it to the front with the method focus() method without having to recreate the window
    if(this.globalVars.popup != null && !this.globalVars.popup.closed){
      this.globalVars.popup.focus();
      return;
    }

    var that = this;

    var url = M.cfg.wwwroot;
    url += "/lib/editor/atto/plugins/reciteditor/view.php";

    this.globalVars.popup = window.open(url, 'HTML Bootstrap Editor', 'scrollbars=1');

    if (this.globalVars.popup.outerWidth < screen.availWidth || this.globalVars.popup.outerHeight < screen.availHeight)
    {
      this.globalVars.popup.moveTo(0,0);
      this.globalVars.popup.resizeTo(screen.availWidth, screen.availHeight);
    }
  }
});
```

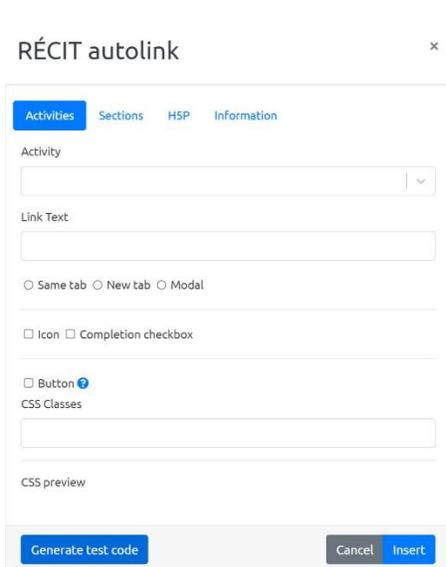
Figure 4 – Dans un plugiciel Atto, nous devons créer un popup dans le YUI JS comme cela.

## Générer le zip d'installation

Lorsque votre plugiciel est prêt à être mis en production, il est important d'enlever les fichiers de cache React, sinon votre dossier zip sera très volumineux. Il est nécessaire d'enlever le `.cache`, `node_modules` et `package-lock.json`. Vous pouvez aussi générer votre zip à l'aide de la commande suivante (vous devez avoir 7zip) :

```
"c:\Program Files\7-Zip\7z.exe" a -mx "pluginname.zip" "src\*" -mx0 -xr!"src\react\.cache" -xr!"src\react\node_modules" -xr!"src\react\package-lock.json"
```

# Des résultats obtenus



## R CIT Autolink

Le logiciel [atto\\_recitautolink](#) facilite l'utilisation du logiciel [moodle-filter\\_recitautolink](#). Il peut  tre utilis  de diff rentes mani res :

- Pour cr er des liens vers des sections ou des activit es
- Pour appliquer des classes CS
- Pour g n rer un bouton Bootstrap
- Pour g n rer des informations sur l'utilisateur
- Pour int grer H5P de la banque de contenu

Figure 5 –  
L'interface d velopp e en React JS  
pour un logiciel de type Atto.

##  diteur HTML Bootstrap R CIT

L' diteur [HTML Bootstrap R CIT](#) vous permet de cr er une mise en page beaucoup plus attrayante et *responsive*. Il est bas  sur le *framework* Bootstrap 4 et son utilisation par glisser-d poser permet d'int grer facilement du contenu dans Moodle. Il offre une large gamme d' l ments (texte, image, vid o YouTube,  l ments d'int gration, etc.) couramment utilis s dans les pages web actuelles.

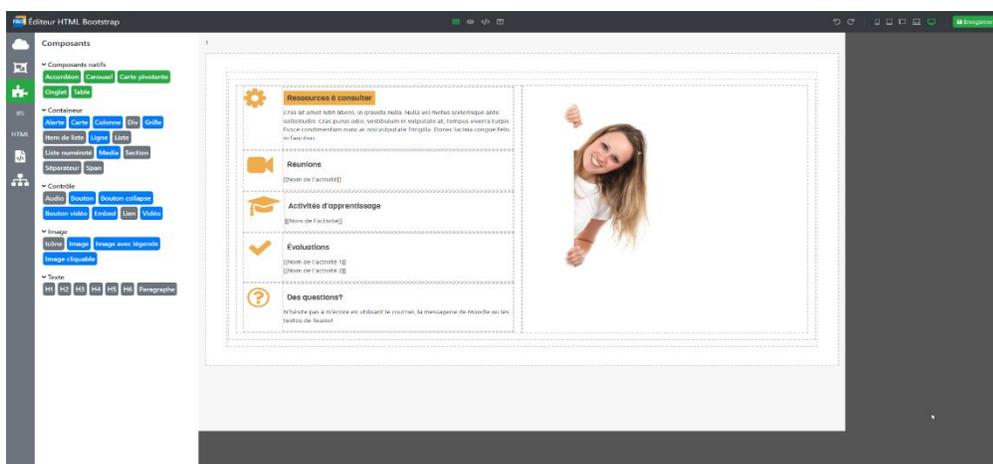


Figure 6 –  
L'interface d velopp e en React JS qui impl mente un  diteur HTML Bootstrap.

# Apprentimètre RÉCIT

L'Apprentimètre RÉCIT organise les données et les présente à l'enseignant de manière à faciliter le suivi des apprenants et la gestion de la classe. Son tableau de bord propose à l'enseignant des alertes et des rapports sur l'assiduité et la performance des apprenants.

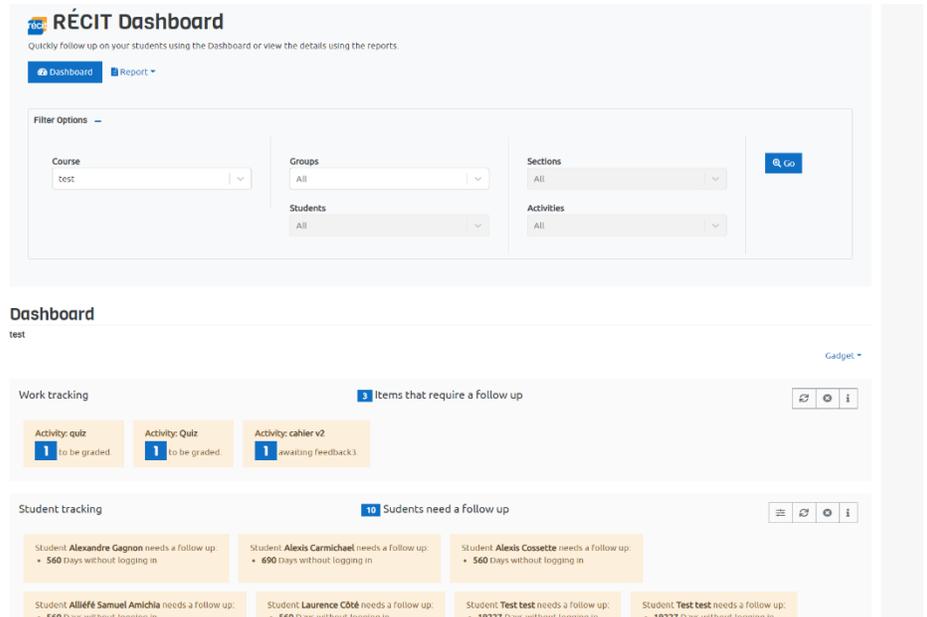


Figure 7 – L'interface développée en React JS pour un plugiciel de type local.

## Cahier de traces

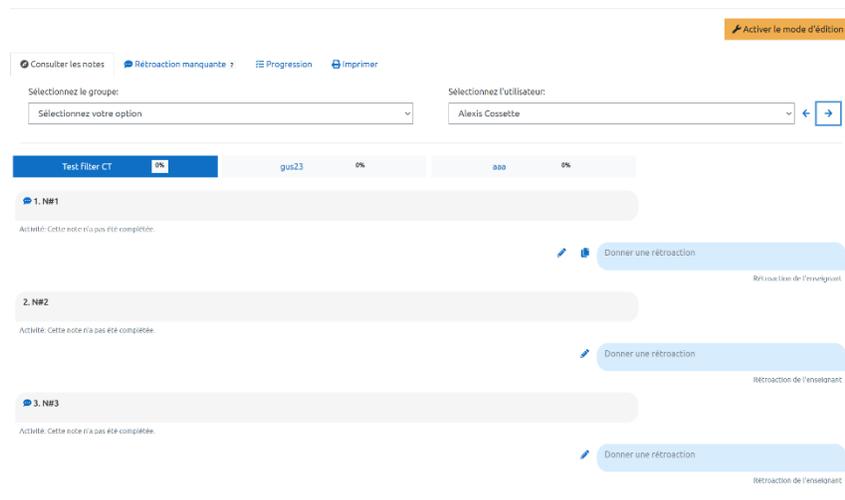


Figure 8 – L'interface développée en React JS pour un plugiciel de type module.

## Cahier de traces

Le cahier des traces développé par le RÉCIT permet à l'apprenant d'écrire ses notes de cours à l'intérieur des ressources et activités offertes dans Moodle. Les notes sont imbriquées à travers les contenus à la manière d'un cahier d'apprentissage.

# Conclusion

L'intégration de React JS à Moodle a rendu possible la réalisation de plusieurs projets numériques qui visent à améliorer l'éducation, que ce soit pour les enseignants, pour les apprenants et pour les gestionnaires.

En terminant, nous désirons mettre en relief le fait que l'Éditeur HTML Bootstrap créée par l'équipe du Service national du RÉCIT de la formation à distance a remporté le deuxième prix lors de l'hackathon de l'Organisation des Nations unies dans le volet « [Branchez-vous sur l'éducation](#) ». Sans l'intégration de React JS à Moodle, le développement de cet outil par notre équipe aurait été difficilement réalisable.

De nouveaux projets numériques en éducation faisant appel à React JS sont présentement dans le collimateur du RÉCIT FAD, comme l'utilisation de la réalité virtuelle dans Moodle...

# Références

1. <https://moodle.org/>
2. <https://fr.reactjs.org/>
3. <https://github.com/SN-RECIT-formation-a-distance/>
4. <https://unite.un.org/news/winners-%E2%80%9Cget-plugged-education%E2%80%9D-hackathon>