

Note de stage 2022

⌘UH Group⌘

Semaine n° 1

⌘Objectifs

Contexte : UH Lawyers¹ est une application créée pour les avocats, un Add-In Word vient la compléter en proposant l'insertion facile de paragraphes pré-écrits dans un document Word ainsi que des « champs de fusion » qui permettront à l'avocat de positionner des sortes de « placeholder » qui seront automatiquement remplis par l'application UH Lawyers grâce aux fonctionnalités de PHPWord.

1. Reprendre le projet sur l'Add-In Word.
2. Se renseigner sur PHPWord² et tester³.
3. Réaliser les fonctionnalités de l'Add-in avec des données provenant d'une API.
4. Se renseigner sur comment convertir un fichier Word en PDF avec PHP.

¹ [Showcase LinkedIn de UH Lawyers](#)

² [Documentation générale sur PHPWord](#)

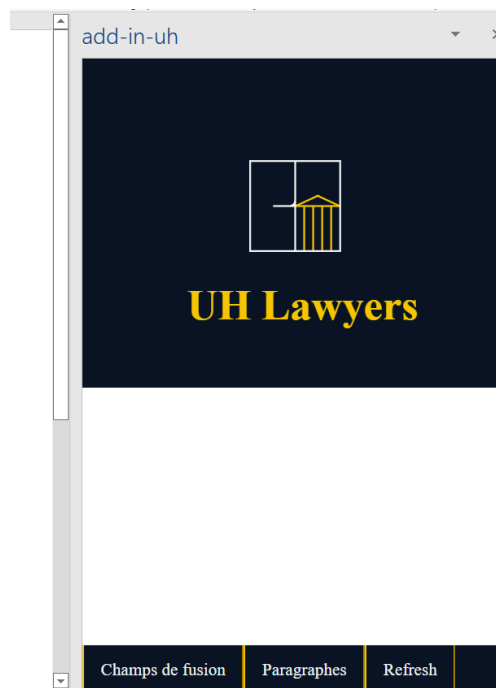
³ [Documentation sur l'installation de PHPWord](#)

Réalisations

Pour cette reprise de projet, j'ai commencé par me réapproprier le projet que j'avais fait précédemment. J'ai donc de nouveau consulté le tutoriel de Microsoft⁴, les fichiers du projet et mes rapports de l'année précédente. Pour repartir sur de bonnes bases j'ai décidé de faire un nouveau projet composé de ce que j'avais déjà (de plus les anciens projets possèdent des erreurs à cause de mes différents tests). Avec l'expérience de l'année dernière j'ai rencontré beaucoup moins de difficulté et d'erreurs lors de la mise en place du projet, et lorsque l'Add-In plantait je savais pourquoi et cela m'a permis de me corriger plus facilement.

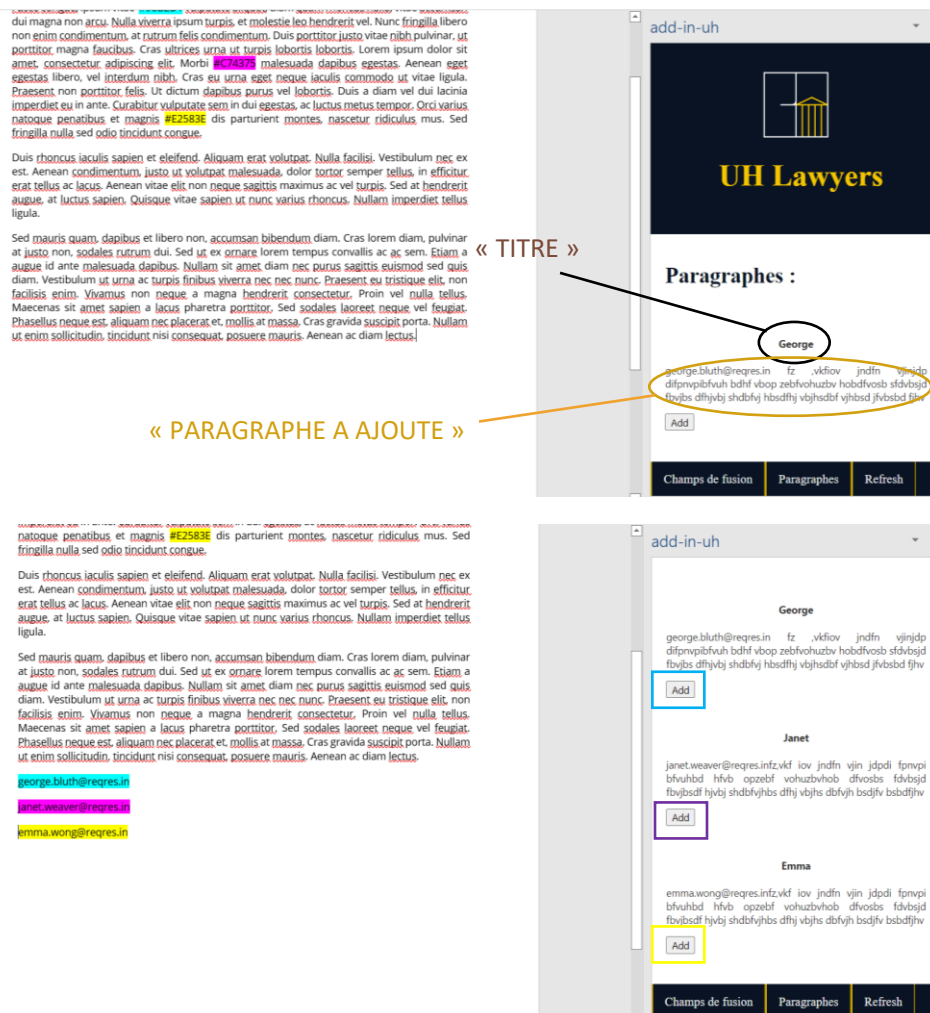
J'ai aussi dû faire quelques révisions sur le JavaScript : 'while', 'for', 'foreach', 'if', 'array', 'innerHTML', 'onclick'.

Ainsi sur ce nouveau projet j'ai commencé par faire la mise en forme de l'application. Faire en sorte que le bandeau soit personnalisé pour l'application UH Lawyers ainsi qu'une barre de navigation permettant de faire apparaître les différents onglets de fonctionnalité :



⁴ [Tutoriel Microsoft Add-In Word](#)

Ensuite j'ai implémenté les fonctionnalités que m'a demandé mon maître de stage. J'ai commencé par la fonctionnalité « Paragraphes », j'ai eu pas mal de difficultés car avec les différentes sources que j'ai consultées me rendait assez confuse sur la syntaxe à utiliser. Notamment parce que j'étais partie sur la syntaxe avec les « Littéraux de gabarits »⁵ pour l'interprétation des données récupérées par Axios⁶ alors que cela n'était pas nécessaire et restrictif.



Lorsque l'on clique sur le bouton « Add » le paragraphe concerné s'ajoute automatiquement à la fin du document (si besoin ce paramètre peut être changé dans le futur).

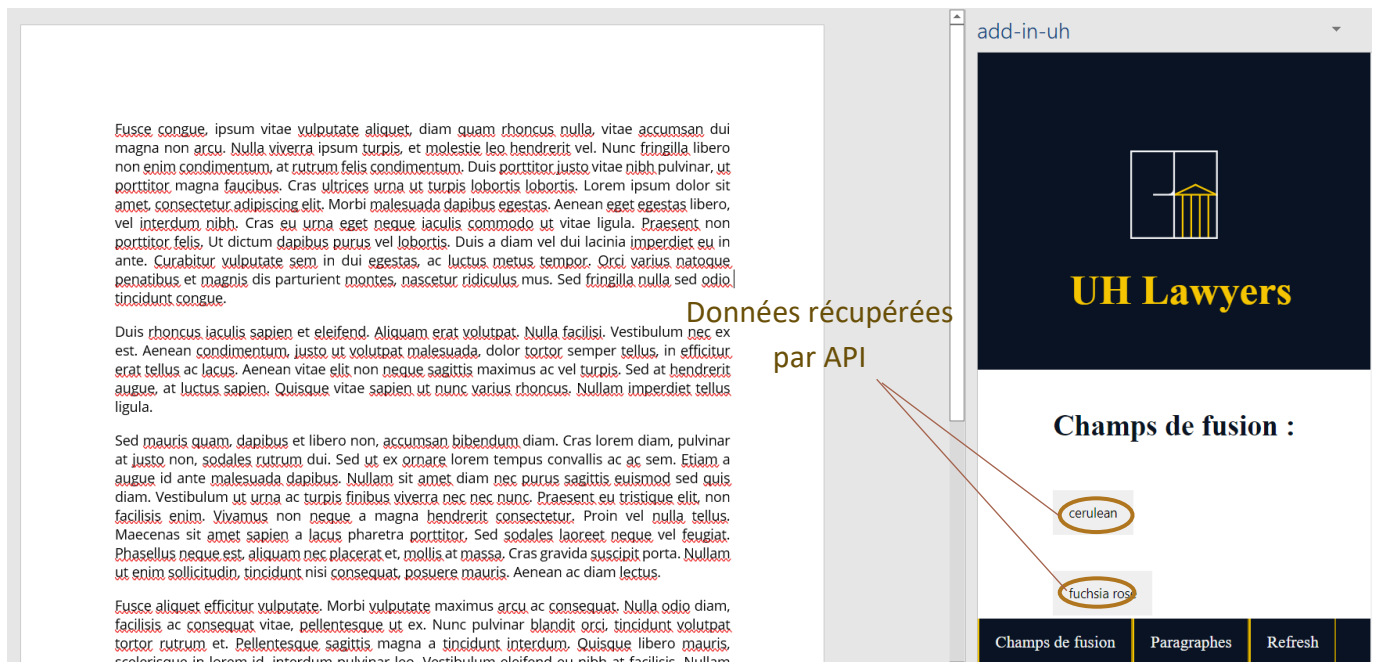
Le paragraphe sur la droite ne correspond pas au paragraphe sur le document car c'est seulement les données de l'API que j'ajoute au document, le texte que j'ai rajouté était un moyen pour moi d'améliorer le visuel de l'application.

⁵ [Documentation MDN Littéraux de gabarits](#)

⁶ [GitHub Axios](#) & [Documentation Axios](#)

A l'origine j'avais créé une API de test avec des paragraphes en base de données, mais malheureusement en voulant l'essayer je me suis rappelé que l'Add-In Word n'accepte pas les APIs avec un protocole HTTP, seulement celles avec un protocole HTTPS. Après quelques recherches il semble possible de changer le protocole de mon API en HTTPS grâce à l'outil « mckert »⁷. J'ai cependant préféré utiliser une API de reqres.in que j'avais déjà utilisé pour pouvoir gagner du temps et progresser sur la partie fonctionnelle de l'Add-In (Je n'ai malheureusement pas réussi à trouver une API en HTTPS contenant plusieurs données qui collait mieux au contexte).

De la même manière, je me suis occupé de la fonctionnalité « Champs de fusion ». Au début ce que je devais faire dans cette fonctionnalité était très flou pour moi, mon maître de stage utilisait les termes de « fusion », « champs de fusion », « qui permettra de fusionner le document une fois que l'utilisateur l'importera dans l'application » ... Ce qui m'a mené à différentes pistes avec l'existence de champs de fusion natifs à Word et différentes fonctions de fusion (« merge ») proposées par PHPWord. Finalement après clarification et quelques recherches il fallait utiliser les « Templates⁸ » de PHPWord qui permettent d'ajouter des sortes de « Placeholder » au document qui seront remplis automatiquement par l'application UH Lawyers (et ici de l'application de test que j'ai créée).



⁷ [GitHub mckert](#)

⁸ [Documentation Templates PHPWord](#)

En condition réelle on aurait `{cerulean}`, ici c'est parce que la donnée provient d'une API quelconque

Fusce congue, ipsum vitae `#98B2D1` vulputate aliquet, diam quam rhoncus nulla, vitae accumsan dui magna non arcu. Nulla viverra ipsum turpis, et molestie leo hendrerit vel. Nunc fringilla libero non enim condimentum, at rutrum felis condimentum. Duis porttitor justo vitae nibh pulvinar, ut porttitor magna faucibus. Cras ultrices urna ut turpis lobortis lobortis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi `#C74375` malesuada dapibus egestas. Aenean eget egestas libero, vel interdum nibh. Cras eu urna eget neque laculis commodo ut vitae ligula. Praesent non porttitor felis. Ut dictum dapibus purus vel lobortis. Duis a diam vel dui lacinia imperdiet eu in ante. Curabitur vulputate sem in dui egestas, ac luctus metus tempor. Orci varius natoque penatibus et magnis `#E2583E` dis parturient montes, nascetur ridiculus mus. Sed fringilla nulla sed odio tincidunt congue.

Duis rhoncus laculis sapien et eleifend. Aliquam erat volutpat. Nulla facilisi. Vestibulum nec ex est. Aenean condimentum, justo ut volutpat malesuada, dolor tortor semper tellus, in efficitur erat tellus ac lacus. Aenean vitae elit non neque sagittis maximus ac vel turpis. Sed at hendrerit augue, at luctus sapien. Quisque vitae sapien ut nunc varius rhoncus. Nullam imperdiet tellus ligula.

Sed mauris quam, dapibus et libero non, accumsan bibendum diam. Cras lorem diam, pulvinar at justo non, sodales rutrum dui. Sed ut ex ornare lorem tempus convallis ac ac sem. Etiam a augue id ante malesuada dapibus. Nullam sit amet diam nec purus sagittis euismod sed quis diam. Vestibulum ut urna ac turpis finibus viverra nec nec nunc. Praesent eu tristique elit, non facilisis enim. Vivamus non neque a magna hendrerit consectetur, Proin vel nulla tellus. Maecenas sit amet sapien a lacus pharetra porttitor. Sed sodales laoreet neque vel feugiat. Phasellus neque est, aliquam nec placerat et, mollis at massa. Cras gravida suscipit porta. Nullam ut enim sollicitudin, tincidunt nisi consequat, posuere mauris. Aenean ac diam lectus.

Fusce aliquet efficitur vulputate. Morbi vulputate maximus arcu ac consequat. Nulla odio diam, facilisis ac consequat vitae, pellentesque ut ex. Nunc pulvinar blandit orci, tincidunt volutpat tortor rutrum et. Pellentesque sagittis magna a tincidunt interdum. Quisque libero mauris, scelerisque in lorem id, interdum pulvinar leo. Vestibulum eleifend eu nibh at facilisis. Nullam

add-in-uh

Champs de fusion :

cerulean

fuchsia rose

true red

aqua sky

tigerlily

blue turquoise

Champs de fusion

Paragraphes

Refresh

Quant au bouton « Refresh » il sert à rafraîchir les données en provenance de l'API.

Pour ces fonctionnalités j'ai travaillé avec l'API proposé par reqres.in, la « users »⁹ pour les paragraphes et la « unknown »¹⁰ pour les champs de fusion afin de varier un peu les données :

```

reqres.in/api/users

{"page":1,"per_page":6,"total":12,"total_pages":2,"data":
[{"id":1,"email":"george.bluth@reqres.in","first_name":"George","last_name":"Bluth","avatar":"https://reqres.in/img/faces/1-image.jpg"},
{"id":2,"email":"janet.weaver@reqres.in","first_name":"Janet","last_name":"Weaver","avatar":"https://reqres.in/img/faces/2-image.jpg"},
{"id":3,"email":"emma.wong@reqres.in","first_name":"Emma","last_name":"Wong","avatar":"https://reqres.in/img/faces/3-image.jpg"},
{"id":4,"email":"eve.holt@reqres.in","first_name":"Eve","last_name":"Holt","avatar":"https://reqres.in/img/faces/4-image.jpg"},
{"id":5,"email":"charles.morris@reqres.in","first_name":"Charles","last_name":"Morris","avatar":"https://reqres.in/img/faces/5-image.jpg"},
{"id":6,"email":"tracey.ramos@reqres.in","first_name":"Tracey","last_name":"Ramos","avatar":"https://reqres.in/img/faces/6-image.jpg"}],
"support":{"url":"https://reqres.in/#support-heading","text":"To keep ReqRes free, contributions towards server costs are appreciated!"}}

reqres.in/api/unknown

{"page":1,"per_page":6,"total":12,"total_pages":2,"data":
[{"id":1,"name":"cerulean","year":2000,"color":"#98B2D1","pantone_value":"15-4020"},
{"id":2,"name":"fuchsia rose","year":2001,"color":"#C74375","pantone_value":"17-2031"},
{"id":3,"name":"true red","year":2002,"color":"#BF1932","pantone_value":"19-1664"},
{"id":4,"name":"aqua sky","year":2003,"color":"#7BC4C4","pantone_value":"14-4811"},
{"id":5,"name":"tigerlily","year":2004,"color":"#E2583E","pantone_value":"17-1456"},
{"id":6,"name":"blue turquoise","year":2005,"color":"#53B0AE","pantone_value":"15-5217"}],
"support":{"url":"https://reqres.in/#support-heading","text":"To keep ReqRes free, contributions towards server costs are appreciated!"}}

```

⁹ Lien API users

¹⁰ Lien API Unknown

Et donc voici le code de l'Add-In Word :

- HTML

```

src > taskpane > taskpane.html > html > body.ms-font-m.ms-welcome.ms-fabric > main#app-body.ms-welcome_main > section#champs-fusion > div.div-contenu > div#champs
13
14 <!-- Office JavaScript API -->
15 <script type="text/javascript" src="https://appsforoffice.microsoft.com/lib/1.1/hosted/office.js"></script>
16 <script src="taskpane.js"></script>
17
18 <!-- CDNs Axios, Bootstrap -->
19 <script src="https://cdn.jsdelivr.net/npm/axios@0.24.0/axios.js" integrity="sha512-RT3I3u0H2waemM8ccCU1PMdU0n8dJCH46N3H2uZ0Y7awMn1Yn7c565sE2UBMppjndeZ91hm87TP1eU6MhJQ="
20 <!-- <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4KWBq78IYHf1dVkuhFTAU6auUBtT94W/HftjDbrCEXSU1oBoqyl2
21 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4gmtz2MlQnik1lWgYs0g+OQMhuP+IIRH9sEMBO0LRn5q+8nbTov4+1p" crosss
22
23 <!-- For more information on Fluent UI, visit https://developer.microsoft.com/fluentui#/. -->
24 <link rel="stylesheet" href="https://static2.sharepointonline.com/files/fabric/office-ui-fabric-core/9.6.1/css/fabric.min.css"/>
25
26 <!-- Template styles -->
27 <link href="taskpane.css" rel="stylesheet" type="text/css" />
28 </head>
29
30 <body class="ms-font-m ms-welcome ms-fabric">
31 <header class="ms-welcome_header ms-bgColor-neutrallighter">
32 
33 <h1 class="ms-font-su">UH Lawyers</h1>
34 </header>
35 <section id="sideload-msg" class="ms-welcome_main">
36 <h2 class="ms-font-xl">Please sideload your add-in to see app body.</h2>
37 </section>
38 <main id="app-body" class="ms-welcome_main">
39 <section id="champs-fusion">
40 <div class="div-contenu">
41 <div id="champs">
42
43
44 <div>
45 <!-- <button class="ms-Button" id="insert-champ">Insert champ</button><br><br>
46 <button class="ms-Button" id="insert-paragraph">Insert Paragraph</button><br><br>
47 <button class="ms-Button" id="insert-paragraph-api">Insert Paragraph API</button><br><br>
48 <button class="ms-Button" id="insert-paragraph-api-test">Insert Paragraph API test</button><br><br><br><br><br><br><br> -->
49 </div>
50 </section>
51 <section id="paragraphes">
52 <h3>Paragraphes : </h3><br>
53 <div class="div-contenu">
54 <div id="paraphs">
55
56 </div>
57 </div>
58 </section>
59 </main>

```

```

EXPLORER
taskpane.html 3
manifest.xml
webpack.config.js
taskpane.css
commands.html
taskpane.js
taskpane.js (Working)

OPEN EDITORS
taskpane.html 3
manifest.xml
webpack.config.js
taskpane.css
commands.html
taskpane.js
taskpane.js (Working)

src > taskpane > taskpane.html > html > body.ms-font-m.ms-welcome.ms-fabric > main#app-body.ms-welcome_main > section#champs-fusion > div.div-contenu > div#champs
55 </div>
56 </div>
57 </section>
58
59 </main>
60 <section class="app-navigation">
61 <nav class="navbar">
62 <ul class="nav-links">
63 <li>
64 <button id="cfb" class="nav-item" type="button"> Champs de fusion </button>
65 </li>
66 <li>
67 <button id="pb" class="nav-item" type="button"> Paragraphes </button>
68 </li>
69 <li>
70 <button id="ref" class="nav-item" type="button"> Refresh </button>
71 </li>
72 </ul>
73 </nav>
74 </section>
75 </body>
76
77 </html>
78

```

• JS

```

OPEN EDITORS
taskpane.ht... 3
manifest.xml
webpack.config.js
taskpane.css src\...
commands.html...
taskpane.js src\...
ADD-IN...
.vscode
extensions.json
launch.json
settings.json
tasks.json
assets
icon-16.png
icon-32.png
icon-64.png
icon-80.png
icon-128.png
logo-filled.png
logo.svg
node_modules
src
commands
commands.html
commands.js
taskpane
taskpane.css
taskpane... 3
taskpane.js
.eslintrc.json
.gitignore
babel.config.json
manifest.xml
package-lock.json
package.json
tsconfig.json
webpack.config.js

src > taskpane > taskpane.js > showOrHideCF
You, 9 hours ago | 1 author (You)

1  /*
2  * Copyright (c) Microsoft Corporation. All rights reserved. Licensed under the MIT license.
3  * See LICENSE in the project root for license information.
4  */
5
6  // Data
7  var data;
8  var dataTest;
9  var dataChamps;
10
11  /* global document, Office, Word */
12
13  Office.onReady((info) => {
14    if (info.host === Office.HostType.Word) {
15      // Determine if the user's version of Office supports all the Office.js APIs that are used in the tutorial.
16      if (!Office.context.requirements.isSetSupported('WordApi', '1.3')) {
17        console.log('Sorry. The tutorial add-in uses Word.js APIs that are not available in your version of Office.');

```

```

OPEN EDITORS
taskpane.ht... 3
manifest.xml
webpack.config.js
taskpane.css src\...
commands.html...
taskpane.js src\...
ADD-IN...
.vscode
extensions.json
launch.json
settings.json
tasks.json
assets
icon-16.png
icon-32.png
icon-64.png
icon-80.png
icon-128.png
logo-filled.png
logo.svg
node_modules
src
commands
commands.html
commands.js
taskpane
taskpane.css
taskpane... 3
taskpane.js
.eslintrc.json
.gitignore
babel.config.json
manifest.xml
package-lock.json
package.json
tsconfig.json
webpack.config.js

src > taskpane > taskpane.js > showOrHideCF
41
42  // Tutoriel & Tests
43
44  function insertParagraph() {
45    Word.run(function (context) {
46
47      var docBody = context.document.body;
48      docBody.insertParagraph("Office has several versions, including Office 2016, Microsoft 365 subscription, and Office on the web.", "Start");
49
50      return context.sync();
51    })
52    .catch(function (error) {
53      console.log("Error: " + error);
54      if (error instanceof OfficeExtension.Error) {
55        console.log("Debug info: " + JSON.stringify(error.debugInfo));
56      }
57    });
58  }
59
60  function insertParagraphAPI() {
61    Word.run(function (context) {
62      if (!data) {
63        fetchData();
64        var docBody = context.document.body;
65        docBody.insertParagraph(
66          `${data.data.data[0].email}`, "Start"
67        );
68      }
69      else {
70        var docBody = context.document.body;
71        docBody.insertParagraph(
72          `${data.data.data[0].email}`, "Start"
73        );
74      }
75      return context.sync();
76    })
77    .catch(function (error) {
78      console.log("Error: " + error);
79      if (error instanceof OfficeExtension.Error) {
80        console.log("Debug info: " + JSON.stringify(error.debugInfo));
81      }
82    });
83  }
84

```



```

src > taskpane > taskpane.js > showOrHideCF
85 function insertParagraphApiTest() {
86     Word.run(function (context) {
87         if(!dataTest) {
88             fetchDataApiTest();
89             var docBody = context.document.body;
90             docBody.insertParagraph(
91                 `${dataTest[0].title}`, "Start"
92             );
93         }
94         else {
95             var docBody = context.document.body;
96             docBody.insertParagraph(
97                 `${dataTest[0].title}`, "Start"
98             );
99         }
100     }
101     return context.sync();
102 })
103 .catch(function (error) {
104     console.log("Error: " + error);
105     if (error instanceof OfficeExtension.Error) {
106         console.log("Debug info: " + JSON.stringify(error.debugInfo));
107     }
108 });
109 }
110
111 function insertChamp() {
112     fetchDataApiTestChamps();
113     Word.run(function (context) {
114
115         var doc = context.document;
116         var originalRange = doc.getSelection();
117         originalRange.insertText( dataChamps[2].name , "End");
118
119         originalRange.load("text");
120         return context.sync();
121     })
122     .catch(function (error) {
123         console.log("Error: " + error);
124         if (error instanceof OfficeExtension.Error) {
125             console.log("Debug info: " + JSON.stringify(error.debugInfo));
126         }
127     });
128 }
129
130
131 // Fonction application finale
132

```

```

src > taskpane > taskpane.js > showOrHideCF
130
131 // Fonction application finale
132
133 function showOrHideCF() {
134     fetchDataApiTestChamps();
135     var contenu = document.getElementById('champs');
136
137     var name = "" + dataChamps[0].name + "";
138     contenu.innerHTML = '<button class="ms-Button" onclick="addChamp('+ name +')"> ' + dataChamps[0].name + ' </button><br><br>';
139     // contenu.innerHTML = '<button class="champ-button" onclick="insertChamp()"> ' + dataChamps[0].name + ' </button>';
140     for (let i = 1; i < dataChamps.length ; i++) {
141         name = "" + dataChamps[i].name + "";
142         contenu.innerHTML += '<button class="ms-Button" onclick="addChamp('+ name +')"> ' + dataChamps[i].name + ' </button><br><br>';
143         // contenu.innerHTML += '<button class="champ-button" onclick="insertChamp()"> ' + dataChamps[i].name + ' </button>';
144     }
145
146     var champ = document.getElementById('champs-fusion');
147     var para = document.getElementById('paragraphes');
148     if(champ.style.display == "none") {
149         champ.style.display = "block";
150         para.style.display = "none";
151     }
152     else {
153         champ.style.display = "none";
154     }
155 }
156

```



```

src > taskpane > taskpane.js > showOrHideP
157 function showOrHideP() {
158   fetchDataApiTest();
159 }
160
161 var contenu = document.getElementById('paragraphs');
162
163 //var nb = parseInt(`${dataTest.data.data[0].length}`);
164 //var nb = Object.keys(`${dataTest.data.data[0].length}`);
165 var title = dataTest[0].first_name;
166 var description = dataTest[0].email + " fz ,vkflov jndfn yjindp dfgmuplfrun bnf vop ietfvonuzbv hndfvosb stndbcjz fvdjhs dfrjdud chufvjv hnddfrj vjhdvdf vhdvd jvdsbd fjhv";
167 contenu.innerHTML += `<div class="text-align:center"><div class="description">+ description +</div><div class="add-button" onclick="addParagraph(0)">Add</div></div>`;
168 for (let i = 1; i < dataTest.length; i++) {
169   var title = dataTest[i].first_name;
170   var description = dataTest[i].email + " fz,vkf iou jndfn yjindp dfgmuplfrun bnf vop ietfvonuzbv hndfvosb stndbcjz fvdjhs dfrjdud chufvjv hnddfrj vjhdvdf vhdvd jvdsbd fjhv";
171   contenu.innerHTML += `<div class="text-align:center"><div class="description">+ description +</div><div class="add-button" onclick="addParagraph(+ 1 +)">Add</div></div>`;
172 }
173
174 var champ = document.getElementById('champs-fusion');
175 var para = document.getElementById('paragraphs');
176 if(para.style.display == "none") {
177   para.style.display = "block";
178   champ.style.display = "none";
179 }
180 else {
181   para.style.display = "none";
182 }
183
184 function addParagraph(ind){
185   fetchDataApiTest();
186   Word.run(function (context) {
187     var docBody = context.document.body;
188     docBody.insertParagraph(dataTest[ind].email,"end");
189   })
190   return context.sync();
191 }
192
193 .catch(function (error) {
194   console.log("Error: " + error);
195   if (error instanceof OfficeExtension.Error) {
196     console.log("Debug info: " + JSON.stringify(error.debugInfo));
197   }
198 });
199 }

```

```

src > taskpane > taskpane.js > showOrHideP
200 function addChamp(nom) {
201   fetchDataApiTestChamps();
202   Word.run(function (context) {
203     let i = 0;
204     while( dataChamps[i].name != nom ){
205       i++;
206     }
207     var doc = context.document;
208     var originalRange = doc.getSelection();
209     originalRange.insertText( " " + dataChamps[i].color , "End");
210     originalRange.load("text");
211
212     return context.sync();
213   })
214   .catch(function (error) {
215     console.log("Error: " + error);
216     if (error instanceof OfficeExtension.Error) {
217       console.log("Debug info: " + JSON.stringify(error.debugInfo));
218     }
219   });
220 }
221
222
223 function fetchData() {
224   axios.get('https://reqres.in/api/users')
225     .then(resp => {
226       // handle success
227       showOutput(resp);
228     })
229     .catch(error => {
230       // handle error
231       console.error(error);
232     })
233 }
234
235 function showOutput(resp) {
236   data = resp;
237   // data = `${resp.data.data[0].email}`;
238 }
239
240 function fetchDataApiTest() {
241   axios.get('https://reqres.in/api/users')
242     .then(resp => {
243       // handle success
244       dataTest = resp.data.data;
245     })
246     .catch(error => {
247       // handle error
248       console.error(error);
249     })
250 }
251 }

```

```

src > taskpane > taskpane.js > ...
251
252
253 function fetchDataApiTestChamps() {
254     axios.get('https://reqres.in/api/unknown')
255     .then(resp => {
256         // handle success
257         dataChamps = resp.data.data;
258     })
259     .catch(error => {
260         // handle error
261         console.error(error);
262     })
263 }
264
265
266 function refresh() {
267     fetchData();
268 }

```

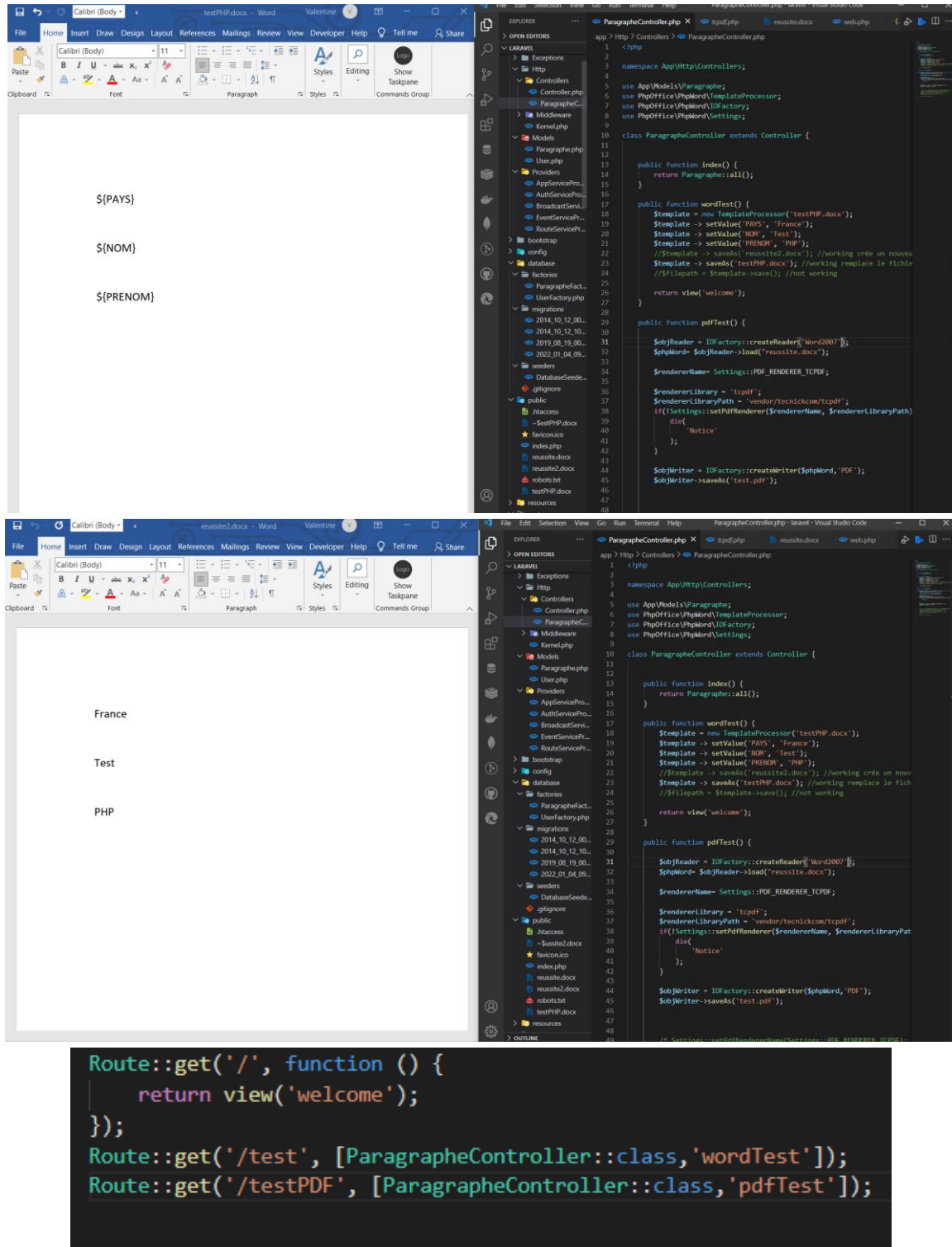
- CSS

```

src > taskpane > taskpane.css > .add-button
107 h1 {
108     font-size: 54px;
109     padding-left: 3.76%;
110 }
111
112 h3 {
113     color: #091323;
114     font-size: 30px;
115     font-family: 'Sen';
116     font-weight: 700;
117     line-height: normal;
118 }
119
120 #app-bosy {
121     padding-bottom: 20px;
122 }
123
124 .app-navigation {
125     background-color: #091323;
126     position: fixed;
127     bottom: 0;
128     width: 100%;
129 }
130
131 #paragraphes {
132     width: 100%;
133     display: block;
134 }
135
136 .navbar {
137     width: 100%;
138     justify-content: space-between;
139     font-family: 'Sen';
140     background-color: #091323;
141     position: fixed;
142     bottom: 0;
143     width: 100%;
144 }
145
146 .nav-link {
147     position: relative;
148     margin-right: 32px;
149 }
150
151 .nav-links {
152     list-style-type: none;
153 }
154
155 .nav-item {
156     float: left;
157     display: block;
158     text-align: center;
159     background-color: #091323;
160     padding: 14px 16px;
161     border-color: #f4c500;
162     color: #f2f4f4;
163     text-decoration: none;
164     font-family: 'Sen';
165     font-size: 17px;
166     border-top-style: none;
167     border-bottom-style: none;
168 }
169
170 #paraphs, #champs {
171     display: block;
172 }
173
174 .div-contenu {
175     float: center;
176     text-align: center;
177     padding-bottom: 100px;
178 }
179
180 .ms-Button {
181     font-family: 'Sen';
182     border-style: none;
183     padding: 14px 16px;
184     display: block;
185 }
186
187 .description {
188     text-align: justify;
189     width: 100%;
190 }
191
192 .add-button {
193     display: block;
194 }
195
196 h4 {
197     display: block;
198 }
199
200 .text-ajout {
201     display: block;
202     padding-bottom: 25px;
203 }

```

En plus de l'Add-In j'ai donc effectué un test avec PHPWord pour les champs de fusion avec un projet Laravel de test :



Actuellement je suis donc sur l'objectif numéro 4, je suis en train de chercher comment convertir les documents Word créés par l'Add-In en PDF une fois importés dans l'application UH Lawyers afin de proposer les deux formats. Pour le moment, cela semble possible via PHPWord plus TCPDF¹¹ ou DOMPDF¹², mais d'après les différents témoignages que j'ai lus cela ne préserverait pas la mise en forme ni les images. Je suis donc toujours sur mes recherches et je consulte les différentes solutions proposées, et je commence à faire des petits tests sur mon projet de test Laravel (peu concluant pour le moment).

Avec les difficultés que j'ai rencontrées et mon inexpérience j'ai consulté pas mal de liens et de blogs avec les mêmes problèmes que moi. J'ai aussi découvert pas mal d'outils qui pourraient me servir dans le futur. Donc j'ai documenté une partie des liens que j'ai consultés : [Document Drive](#).

¹¹ [Site web TCPDF](#) & [GitHub TCPDF](#)

¹² [GitHub TCPDF](#)