



UNIVERSITÉ
DE LORRAINE

LORRAINE
INP



POLYTECH[®]
NANCY

STAGE DE FIN D'ETUDES

Content Manager – Développeur web

Rédigé par : Axel Junior MENGUE FEUKENG

Sous la supervision de :

Jorge CLAUDIO (encadreur professionnel)

Samir YUCEF (encadreur académique)

13/09/2022



**BNP PARIBAS
CARDIF**

L'assureur d'un monde qui change



Sommaire

Table des matières

Sommaire	1
Remerciements	2
Introduction	3
I. Présentation de l'entreprise	4
A. L'entreprise en général	4
1. Présentation de BNP Paribas Cardif.....	4
2. Localisation de BNP Paribas Cardif	7
B. Présentation de la Digital Factory	7
C. Description du poste	8
1. L'environnement de travail	8
2. Les montées en compétences visées	8
II. Déroulement du stage	9
A. Présentation de Liferay	9
B. Gestion du site Finagora.....	10
1. Contribution d'une actualité	11
2. Contribution dans la gamme financière.....	13
C. Gestion du site B2B et B2C.....	16
D. Gestion des différents projets.....	19
1. Projet d'accessibilité numérique	19
2. Projet d'optimisation du temps de formatage.....	25
3. Projet d'intégration d'une page web	36
III. Difficultés rencontrées et acquis du stage	39
A. Difficultés rencontrées	39
B. Les acquis au terme du stage	40
Conclusion.....	41
Annexes.....	42
Annexe 1 : Liens.....	42
Annexe 3 : Table des illustrations.....	43
Annexe 3 : Images	44
Annexe 3 : Table des abréviations.....	46



Remerciements

J'adresse mes sincères remerciements à Polytech Nancy, et particulièrement mon directeur **Eric GNAEDINGER** pour m'avoir fait confiance en m'admettant dans son établissement. L'écoute, l'accompagnement et les enseignements dont j'ai bénéficié m'ont permis de trouver un stage de fin d'études.

Je pense également à madame **Cécilia LAUSTRIAT**, qui a cru en mon potentiel et m'a accueilli(e) au sein de sa direction E-commerce de **BNP Parisbas Cardif**.

De même, je souhaiterais remercier tout particulièrement monsieur **Claudio Jorge** qui m'a épaulé et conseillé. Ainsi que son équipe de Webmaster, notamment monsieur **Jean DOMCHE** et monsieur **Dorian MATSIMA** pour toutes les connaissances pratiques qu'ils m'ont transmises dans le domaine du développement web.

Je remercie également mon tuteur académique monsieur **Samir YUCEF** pour sa disponibilité, ses conseils et son accompagnement.

J'adresse également des remerciements à mon aîné académique **Neil Davy NIE FOTSO** qui m'a été d'une aide précieuse dans la rédaction de mon CV pour la recherche de stage.

Je vais également remercier mon oncle **Jean SEVERIN** pour les conseils de rédaction du rapport de stage qu'il m'a donné.

Je remercie aussi madame **Aurelie SCHMITT** qui m'a donné plusieurs conseils à prendre en compte dans la rédaction d'un rapport.

Grâce à ce stage, j'ai pu voir de près les éléments qui entrent dans la vie d'un site web et de surcroît le site web d'un grand groupe reconnu mondialement. J'ai pu perfectionner mes connaissances en développement en travaillant sur des projets très intéressants et auprès d'une équipe hautement qualifiée.



Introduction

Dans le cadre de la fin de mes études d'ingénieur à Polytech Nancy, je suis tenu d'effectuer un stage de 6 mois en entreprise. Après des mois de recherche, j'ai obtenu 2 propositions ; l'une à Sopra Steria au Luxembourg et l'autre à BNP Paribas Cardif en Ile-de-France. Après étude des 2 propositions, j'ai opté pour BNP Paribas Cardif.

J'ai donc rejoint cette entreprise pour une durée de 23 semaines. J'intègre alors l'équipe de développeurs web de la Digital Factory. À mon arrivée dans l'équipe, mes missions sont les suivantes :

- Participer aux comités éditoriaux, pour obtenir la matière, les informations nécessaires à la création de contenus : textes, vidéos, visuels, podcasts, données contextuelles pour le référencement naturel de Google et données contextuelles pour les réseaux sociaux.
- Intégrer ces contenus web dans l'outil de gestion de contenus.
- Développer des nouvelles pages web maquetées par les équipes de web design et UX/UI.
- Collaborer avec l'équipe d'acquisition dans le but d'ajouter des nouveaux leviers sur le site.
- Assumer le rôle de webmaster dans le but d'intégrer, de modifier ou encore mettre à jour les contenus sur les différents sites web de Cardif à partir du CMS Liferay.
- Intervenir sur les différentes missions et les différents projets que Cardif mène au quotidien.

Je décide de faire ce stage chez BNP Paribas Cardif parce que j'y vois une occasion pour moi de travailler à nouveau sur une interface utilisateur en condition réelle avec les langages de développement web de base que sont le Html, Css et le JavaScript. C'est grâce à ces langages de base que j'ai poursuivi une formation de 5 ans dans le développement web.

L'environnement de travail au sein de Cardif va donc être l'outil de gestion des contenus (Liferay) et un éditeur de texte : Visual studio code dans mon cas. J'ai au long de mon stage travaillé sur plusieurs projets ; Intégration de pages, accessibilité du site Web, Gestion des contenus. Ces projets ont tous été intéressants. J'ai rencontré des difficultés que j'ai résolues. J'ai identifié des problèmes que j'ai résolus ; je pense au projet d'optimisation du temps de formatage des articles en Html.

Dans ce rapport, je vous présente le département E-Business de Cardif et la Digital Factory qui est l'équipe à laquelle je suis rattaché, je présente ensuite le déroulement de mon stage ainsi que les différents projets sur lesquels j'ai travaillé et pour finir les difficultés que j'ai rencontrées d'une part et les acquis du stage d'autre part.



I. Présentation de l'entreprise

A. L'entreprise en général

1. Présentation de BNP Paribas Cardif

« L'assureur d'un monde qui change » est le slogan du leader mondial en assurance des emprunteurs. Cardif du groupe BNP Paribas est leader mondial des partenariats en bancassurance et en assurance emprunteur. En offrant la meilleure assurance à ses clients, BNP Paribas Cardif va donc permettre à ceux-ci de réaliser leurs projets tout en se prémunissant des aléas de la vie tels que : le décès, l'invalidité, l'incapacité de travail et/ou de perte d'emploi. Cardif ne va pas se limiter aux assurances de prêt, elle va créer avec ses partenaires des offres qui correspondent à l'évolution et aux changements dans le monde. Ces offres sont diverses. On pourra noter les secteurs suivants : assurance épargne, prévoyance, retraite, assurance-crédit, assurance emprunteurs, assurance moyens de paiement et bien d'autres. Les principales seront donc l'assurance emprunteur (ADE), l'assurance habitation, assurance prévoyance, épargne et retraite.

Les chiffres clés de Cardif sont les suivants :

Tableau 1 chiffres clés de Cardif (chiffres de 2021)

N° 1	MONDIAL EN ASSURANCE DES EMPRUNTEURS
500	PARTENAIRES DISTRIBUTEURS
8000	COLLABORATEURS DANS LE MONDE
32.6	MD€ DE CHIFFRE D'AFFAIRES EN 2021
33	PAYS

En bâtissant son business model sur le partenariat, l'esprit d'entreprise et l'expertise, BNP Paribas Cardif s'est ainsi créé un partenariat de distribution très fort avec les Conseillers en Gestion de Patrimoine (CGP) et les courtiers. Il leur donne les moyens d'accompagner au mieux leurs clients.

Étant conscient de la place grandissante que prend le digital dans l'activité de ses partenaires et dans la relation qu'ils entretiennent avec leurs clients, BNP Paribas Cardif a fait le choix d'investir fortement dans de nouveaux outils transactionnels tels que le site web des partenaires et des courtiers avec pour ambition d'être la référence en termes de parcours client et partenaire.

BNP Paribas Cardif va se fonder sur la volonté de rendre l'assurance plus accessible. On va donc noter les 5 assurances clés de Cardif ci-dessous :

- Assurer vos crédits avec Cardif Libertés Emprunteur
- Épargner pour vos projets avec Cardif Elite vie et Cardif Elite capitalisation
- Préparer votre retraite avec Cardif Elite Retraite
- Protéger votre famille et/ou votre entreprise avec Cardif Garantie Couverture Personnelle et Cardif Garantie Couverture Entreprise.

- Protéger vos appareils mobiles en cas de casse, vol, utilisation frauduleuse de la ligne de votre mobile, grâce aux offres distribuées par nos partenaires : Orange, Boulanger, Auchan, Mediamarkt, Welcom', Vanden Borre.

Les détails de toutes ces assurances sont retrouvés sur le site web cardif.fr. Il s'agit bien du site web sur lequel je suis amené à travailler durant mon stage. (Source : <https://www.cardif.fr/qui-sommes-nous>)

Ci-dessous, nous pouvons voir l'organigramme de la direction E-Business. Celui-ci liste tous les employés et leurs fonctions en partant de Pascal Perrier qui est mon supérieur n+3.



Figure 1 : Organigramme direction E-Business

À cet organigramme, j'ajoute le second organigramme suivant qui liste les différentes sections au sein même de la E-Business.

Comme nous pouvons le voir sur le second organigramme, Cardif E-Business est formé de 8 équipes :

- **La Team Commerciale** : cette équipe va traiter de façon téléphonique les démarches des clients du site web. C'est pour moi le poste le plus important de la direction car tout travail est fait dans le but d'obtenir des clients qui vont souscrire à l'assurance chez cardif.fr
- **La Team Acquisition** : elle est chargée de récolter le trafic sur le site web, de l'analyser afin de ressortir des estimations, des prédictions et des pistes d'améliorations de certains éléments du site web.

- **La Team réseaux sociaux** : elle s'occupe de la communication externe des produits Cardif.
- **La Team Digital Courtage** : C'est une équipe externe qui nous fournit les contenus pour actualiser les sites partenaires.
- **La Team Édito** : elle se charge de la production des contenus web du site Cardif
- **La Team UX/UI** : elle se charge de maquetter et designer les contenu web de la Team Édito.
- **La Team DSI** : une équipe socle digital qui est responsable des différentes applications qui sont utilisées au sein de Cardif. Elle n'est pas directement rattachée à notre direction.
- **La Team Digital Factory** : constituée des webmasters qui interviennent directement sur l'interface utilisateur des sites web de Cardif. C'est cette équipe que je rejoins pour mon stage de fin d'étude au sein de Cardif

Les Teams DF, Édito, UX/UI, DSI et Digital Courtage sont très souvent amenées à travailler ensemble.

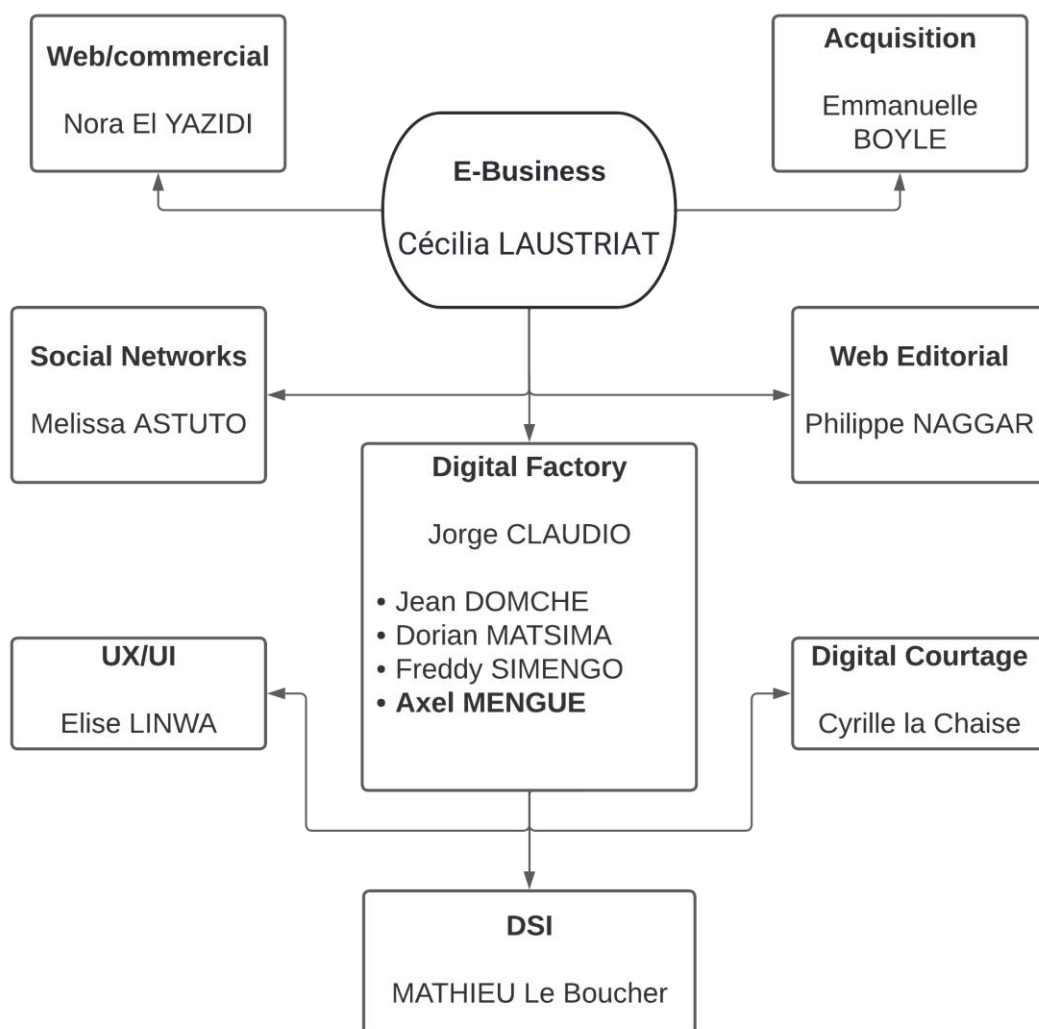


Figure 2 : organigrammes des différentes équipes de la E-Business

2. Localisation de BNP Parisbas Cardif

Cardif se situe aujourd'hui au 8 Rue du Port, 92000 Nanterre. Cette adresse est depuis peu le siège de l'entreprise.

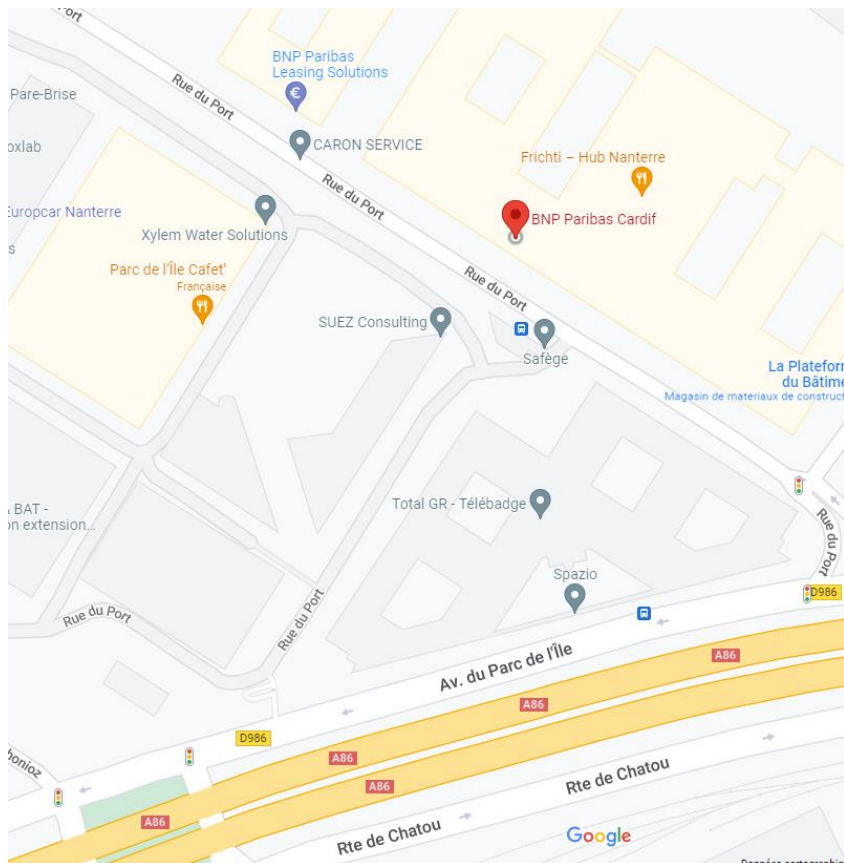


Figure 3 : localisation du siège de BNP Parisbas Cardif

B. Présentation de la Digital Factory

La Digital Factory est l'équipe que j'ai rejoint à mon arrivée à Cardif. Comme je l'ai montré plus haut, c'est une des équipes qui constitue le département E-Business de Cardif.

La DF est essentiellement constituée de développeurs web :

- **Freddy SIMENGO**, il rejoint l'équipe le même jour que moi comme prestataire, avec un profil de développeur front-end.
- **Dorian MATSIMA**, au sein de l'équipe depuis presque 4 ans comme prestataire, il apporte un profil de développeur full stack. Il sera mon premier mentor dans mes débuts
- **Jean DOMCHE**, au sein de la DF depuis près de 8 ans, il m'a été d'une très grande aide pratique dans tous mes projets au sein de l'équipe. Il est donc celui que je considère comme mon n+1 hiérarchique.
- **Jorge CLAUDIO**, qui est le manager de l'équipe. Il a, lui aussi, un profil complet en développement web. C'est donc à lui que je rends des comptes.



- **Axel MENGUE**, je rejoins l'équipe en avril comme stagiaire avec un profil tout aussi complet de développeur full stack.

La DF est donc l'équipe qui s'occupe de la gestion de l'interface utilisateur du site web Cardif. Nos missions vont se résumer à gérer le site web dans son entièreté

C. Description du poste

Le site cardif.fr et le site Finagora des courtiers de Cardif vivent de la création et de la mise à jour des contenus web divers et variés au quotidien. Il peut s'agir des vidéos, images, podcasts ou encore des articles.

Ma mission est de gérer ces sites web à l'aide de l'outil de gestion des contenus au sein de l'entreprise, tout en tenant compte des délais et du planning de publication de l'équipe éditoriale.

Je vais être amené à travailler sur différentes notions digitales pouvant aller du développement web au référencement naturel Google. Je vais tout aussi voir de près la notion d'accessibilité dans un site web et les bonnes pratiques à adopter lors de la réalisation d'une interface utilisateur.

1. L'environnement de travail

J'intègre la Digital Factory qui est une équipe de la direction E-Business comme je l'ai dit plus haut. La B-E est une direction d'équipe aux profils très variés qui travaillent tous pour assurer la vie du site public (B2C) et du site privé (B2B).

Tout au long de mon stage, je serais accompagné par l'équipe de la Digital Factory qui est constitué de 2 webmasters confirmés et du manager qui est mon tuteur.

Je vais être amené à échanger avec plusieurs équipes au cours du stage afin de mener à bien mes missions et de monter en compétences le plus rapidement possible. Je pense notamment à : l'équipe Web éditorial qui rédige les articles, l'équipe data et Acquisition qui surveille le trafic et le référencement naturel des contenus, l'équipe des réseaux sociaux, qui gère les comptes réseaux sociaux de Cardif France et exploite les contenus digitaux dans ses posts. Ces échanges avec des experts digitaux et métiers différents sont riches en apprentissages et donnent du sens au travail réalisé.

2. Les montées en compétences visées

Ce stage me permettra de :

- Découvrir le métier de Content Manager sur le terrain concret de la vie d'un site et dans le cadre d'un projet de migration d'un site,
- Travailler avec tous les profils métiers du digital autour de la vie d'un site : experts en acquisition de trafic, webmasters, chef de projet digital, UX designers, rédacteurs de contenus,
- Apprendre à intégrer des contenus digitaux dans différents outils de gestion de contenus
- Acquérir de nombreuses notions comme le SEO (référencement naturel Google),





- Découvrir un outil de suivi d'audience (Google Analytics) et à interpréter ses tableaux de bord et ses données.

II. Déroulement du stage

Pendant mon stage, j'ai été amené à travailler sur différents univers, plus précisément sur différents sites web. Il s'agit du site cardif.fr qui se divise en deux grands sites : Cardif Particulier et Cardif Partenaire. Puis le site Finagora auquel on accède grâce à un lien sur l'espace Partenaire. Pour toute la suite, on va désigner les différents sites comme suit :

- Cardif Partenaire → B2B (Business to Business)
- Cardif Particulier → B2C (Business to Consumer)
- Finagora → sites des courtiers

Tous ces sites web ont été conçus à partir d'un **Système de Gestion de Contenu** (SGC) en anglais Content Management System (CMS). Un CMS va permettre la création et l'administration d'un site web. Dans notre cas, le CMS que nous utilisons à Cardif est Liferay.

A. Présentation de Liferay

Liferay est un éditeur de logiciels open source permettant à ses clients de créer des portails, intranets et des sites web. Il est tout aussi un éditeur indépendant de solutions de portail open source, de gestion de contenu, de collaboration sociale et de commerce en ligne B2B, en langage Java. La première version de Liferay est apparue en 2004, puis d'autres versions ont vu le jour et aujourd'hui, nous sommes à la version 7.

Pour la suite, nous allons limiter Liferay à sa fonction de CMS. Les fonctionnalités de celui-ci qui vont nous intéresser sont les suivantes :

- Il permet de hiérarchiser les utilisateurs en leur attribuant des rôles, des droits d'accès et/ou des permissions.
- Il permet de structurer les contenus qui peuvent être du contenu web ou des documents et médias.
- Il inclut la gestion des versions.
- Il inclut la planification des publications.
- Il va inclure le référencement SEO

Liferay se compose de 3 principaux produits :

- Liferay Analytics Cloud est un outil d'analyse web
- Liferay Commerce est une solution pour créer des sites web dédiés au B2B.
- Liferay Digital Experience Platform (DXP) qui est ici une plateforme numérique pour la création des portails clients, des intranets et des sites web ouverts au public.

Ce dernier produit comporte deux versions différentes :

- Community Edition (CE) que l'on va désigner comme une version moins stable aux mises à jour, mais elle reste tout de même fonctionnelle.



- Entreprise Edition (EE ou DXP) qui est, comme son nom l'indique, une version faite pour les portails d'entreprises. Elle va être payante, mais néanmoins, elle est plus stable, sécurisé et va garantir la disponibilité des données en cas de bug avéré.

Au sein de Cardif, nous allons utiliser la version Entreprise Edition de Liferay parce qu'elle offre un grand nombre de Portlet, une meilleure sécurité des données pour le passage à la version la plus récente et elle permet également de créer et synchroniser plusieurs sites.

B. Gestion du site Finagora

Finagora est le site web des partenaires et des courtiers en assurance de Cardif. Sur ce site web, les partenaires de Cardif vont trouver une partie de l'actualité de Cardif, mais aussi et surtout les différents éléments nécessaires au bon fonctionnement de leur activité. Ils vont retrouver :

- Les **produits de la gamme Cardif** dont les plus importants sont : Cardif Libertés Emprunteur, Cardif Élite, Cardif Élite Retraite, Cardif Elite Capitalisation...
- L'**actualité de Cardif** et celles des partenaires.
- La **Gamme Financière** qui à son tour est divisée en plusieurs sous parties :
 - **La vie des supports** qui comprend : les Mandats de gestion Cardif Élite, la Gestion pilotée/Gestion Horizon, les Fonds euro et les supports structurés. Nous citons ici les plus importants sur lesquels nous sommes très souvent amenés à travailler.
 - Les **Opérations sur titre**
 - Les **Partenaires Gold** et les **Partenaires Privilégiés**
 - L'**Actualité des partenaires Gold**
 - Les **nouveaux OPC de la gamme Cardif**

Nous allons revenir plus en détail sur les différents points.

- **Juridique et fiscal** où l'on va notamment retrouver l'actualité contenant des points juridiques et tous les documents juridiques.

En tant que membre de la Digital Factory, mon rôle va être de gérer le back office du site web Finagora. Gérer ce site va revenir à : mettre à jour les différents fichiers devenus obsolètes, mettre à jour les actualités et le contenu web de tout le site, publier de nouveaux articles dans les différentes rubriques et section qui les concerne. Pour faire simple, sur Finagora, les éléments que l'on va être appelé à modifier seront soit du contenu web, soit des documents et médias.

Pour la suite, nous allons désigner par **contribution** toute opération visant à modifier le contenu web du site Finagora.

Les différentes missions sur Finagora qui parviennent à la digital DF (Digital Factory) sont confiés par la Digital Courtage.

Comme nous l'avons dit précédemment, la contribution va être centrée sur la création, la modification, la suppression, la dépréciation et la planification des contenus web et des documents & médias. Cette contribution va donc s'appliquer sur les différentes rubriques du

site citées ci-dessus, allant ainsi de la rubrique « produits Cardif » à la rubrique « Juridique et fiscal ».

1. Contribution d'une actualité

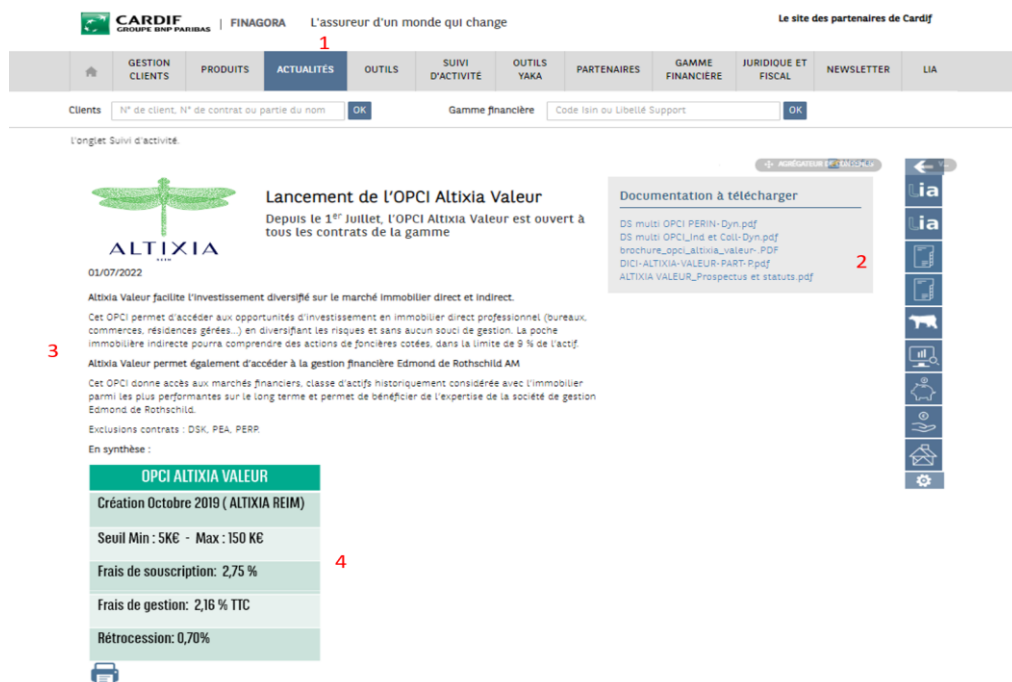
Une actualité est une page web qui aura un contenu web et/ou un onglet avec des documents à télécharger. Le contenu quant à lui aura un titre, un sommaire ou non, un résumé ou non et enfin un corps de l'actualité.

Pour la contribution du contenu web, il faut dans un premier temps formater le contenu de l'actualité en Html. Nous allons prendre en compte ici les entêtes, les paragraphes, la police, les images, les icônes, le format du texte et tout autre élément nécessaire au passage du contenu textuel à une page web. Nous allons particulièrement faire attention aux accents, aux apostrophes et aux points de suspensions, car ceux-ci sont susceptibles de ne pas présenter comme on l'aurait souhaité dans la page web.

Ensuite, il faut prendre en compte les catégorisations d'une part ; en d'autres termes, il faut savoir quelles sont les pages qui ont le droit d'afficher l'actualité ('a la une', 'first page'...), donner les droits d'accès en lecture aux utilisateurs d'autre part.

Exemple de contribution d'une actualité

Je vais ici vous présenter en détail les étapes nécessaires pour contribuer une actualité. Nous allons prendre l'exemple d'une actualité que j'ai mise en ligne : **Altixia**.



The screenshot shows the CARDIF website interface. The top navigation bar includes 'GESTION CLIENTS', 'PRODUITS', 'ACTUALITÉS' (highlighted), 'OUTILS', 'SUIVI D'ACTIVITÉ', 'OUTILS YAKA', 'PARTENAIRES', 'GAMME FINANCIÈRE', 'JURIDIQUE ET FISCAL', 'NEWSLETTER', and 'LIA'. Below the navigation bar, there are search filters for 'Clients' and 'Gamme financière'. The main content area displays an article titled 'Lancement de l'OPCI Altixia Valeur' dated 01/07/2022. The article text describes the investment opportunities and risks. To the right, there is a 'Documentation à télécharger' section with links to various PDF documents. At the bottom of the article, there is a table with financial details.

OPCI ALTIXIA VALEUR	
Création Octobre 2019 (ALTIXIA REIM)	
Seuil Min : 5K€ - Max : 150 K€	
Frais de souscription: 2,75 %	
Frais de gestion: 2,16 % TTC	
Rétrocession: 0,70%	

Figure 4 : actualité Altixia

Sur cet article, on peut facilement remarquer qu'il est une actualité en '1' ; il contient des pièces jointes à télécharger en '2' ; il a un contenu web avec une image en '3' et '4'.

- **Formatage du contenu textuel en code Html.** Dans un premier temps, on va formater le contenu pour obtenir un code Html. Ce qui constitue à identifier les paragraphes, les textes en gras et d'appliquer les balises correspondantes. Pour l'image dans le contenu,

je l'ai stocké dans les documents et médias puis j'ai utilisé son chemin d'accès pour l'afficher dans le contenu web à l'aide de la balise .

- **Préparation du visuel** (voir image ci-dessous).

Étape 1 : le panneau 1 montre tous les onglets nécessaires à la configuration.

Étape 2 : ça correspond à l'onglet contenu de l'étape 1. C'est là qu'on va entrer le résumé de l'article, le contenu formaté précédemment en Html, et si c'est nécessaire, les fichiers à télécharger.

Étape 3 : à ce niveau, on ajoute l'image qui décrit le contenu web ou le logo de l'entreprise.

Étape 4 : comme nous en avons parlé plus haut, c'est à ce niveau que l'on va préciser les rubriques dans lesquelles on veut afficher l'article. En actualité, 'first page' demande l'affiche sur la page d'accueil et sur le page des actualités.

Étape 5 : à cette étape, on a la possibilité de planifier la date et l'heure à laquelle on souhaite mettre l'actualité en ligne. On peut aussi modifier l'heure et la date de mise à jour dans le cas d'une mise à jour de l'article.

Étape 6 : elle définit la mise en ligne dans l'espace restreint de Finagora. Le contenu reste accessible uniquement aux personnes connectées.

Pour résumer, le gros du travail est fait lors du formatage du contenu de l'article proprement dit. Le reste est de la configuration et des paramétrages.

The screenshot shows the 'Lancement de l'OPCI Altixia Valeur' form. The left sidebar (1) contains navigation tabs: Contenu, Résumé (selected), Catégorisation, Programme, Page d'affichage, Contents relatifs, and Champs personnalisés. The main area (2) is the 'Contenu (Requis)' section, which includes a title field, a friendly URL, an 'Accroche (Requis)' field, and a rich text editor for the main content. Below the main content area, there are four panels: 'Résumé' (3) with a 'Sommaire' field and a checkbox for 'Utiliser la petite image' (with a dragonfly logo); 'Catégorisation' (4) with a 'Type' dropdown and 'Actualités' section; 'Programme' (5) with 'Date d'affichage', 'Date d'expiration', and 'Date De Revue' fields; and 'Page d'affichage' (6) with a 'Pages Privées > Actualités > Detail actualité' link and a 'View content in Detail actualité' link.

Figure 5 : étapes de préparation du visuel de publication

2. Contribution dans la gamme financière

Cette rubrique est celle sur laquelle je suis très souvent amené à travailler. Comme nous l'avons expliqué, elle comporte plusieurs sous-sections.

- **La vie des supports** : cette sous rubrique est la plus importante de la gamme financière. Elle comprend les gestions déléguées et sous mandants, l'offre immobilière, les supports structurés et bien d'autres. La contribution visant à modifier le visuel de cette sous-section va se faire à un endroit précis ; dans le contenu web de la vie des supports. Ce contenu est localisé à : Web Content/Accueil/Contribution/Gamme financière/la vie des supports.

Pour la modification des éléments qui la constitue, la procédure sera souvent analogue.

Certains éléments de cette rubrique sont des liens qui mènent au téléchargement d'un document ; c'est le cas de l'offre immobilière. D'autres sont des liens qui pointent vers des pages spécifiques ; c'est le cas des gestions déléguées.

- **Contribution des offres immobilières.** La contribution des docs et Médias se fait à l'emplacement suivant : Accueil/Contribution/Gamme financière/Immobilier. Dans ce dossier, il faudra remplacer les documents obsolètes par les documents fournis par la Digital Courtage.
Dans le cas où l'offre immobilière n'existe pas, il faut la créer. Il va falloir créer son dossier dans les Docs & Médias, y ajouter les documents, modifier le contenu web de la vie des supports afin d'ajouter le lien de cette nouvelle offre et enfin ajouter ses documents en lien (voir image ci-dessous).



Figure 6 : extraits d'offre immobilière de la gamme financière

Les offres SC Pythagore et SCI Cap Santé sont deux des offres que j'ai à créer durant mon stage.

- **Contribution des gestions déléguées.** Il s'agit là de la rubrique qui donne des informations sur les partenaires auprès de qui Cardif s'appuie pour gérer au mieux une partie ou tous les contrats de ses clients. Nous allons retrouver pour tous les partenaires

une actualité et/ou des documents d'information, de présentation et d'évolution du partenaire.

La contribution à ce niveau peut être soit la modification du contenu web d'un partenaire ou l'application des opérations d'ajout, suppression ou de modification des documents du partenaire. La modification du contenu web se fait suivant le nom du partenaire et à l'emplacement ci-contre : Web Content/ Accueil/ Contribution/ Actualités/ Gestion déléguée et sous-mandat Elite. Toujours suivant les noms des partenaires, la contribution des Documents et Médias se fait à l'emplacement ci-contre : Accueil/ Contribution/ Produits épargne/ CARDIF ELITE/ Gestion déléguée et sous mandat (voir image ci-dessous). En 5 mois de stage, j'ai contribué sur les produits de tous les partenaires de Cardif.

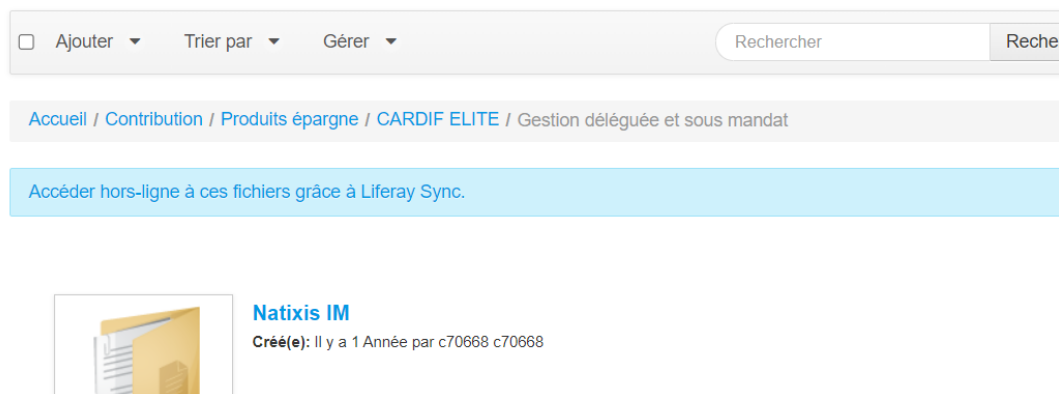


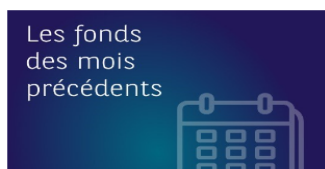
Figure 7: emplacement des documents de gestion délégués

- **Les nouveaux et les anciens fonds** : ils vont de pair. La modification de l'un entraîne très souvent la modification de l'autre. Nous allons décrire ça le plus facilement possible. L'image ci-dessous montre les 'cards' d'entrée dans les rubriques citées à partir de l'onglet 'gamme financière'.



Figure 8 : les fonds de la gamme Cardif.

Il s'agit là de la rubrique où l'on met à jour les fonds des sociétés de gestion partenaire de Cardif. Ces fonds sont ajoutés dans une page web en tant qu'actualité. L'interface des nouveaux fonds ressemble à l'image ci-dessous.



21/01/2022

Date de mise à jour : 10/01/2022

Retrouvez les fonds des sociétés de gestion partenaires du mois précédent.

Les fonds du mois précédent
Retrouvez tous les fonds des mois précédents

Société de gestion	Nom du fonds	Code ISIN
JP Morgan AM	JPM America Equity D (acc) - EUR	Fiche LU0217390490
CPR Asset Management	CPR Invest - Hydrogen - A EUR - Acc	Fiche LU2389405080
Colville Capital Partners France	Silver Autonomie R	Fiche FR00140052N5

Figure 9 : les nouveaux fonds de la gamme Cardif

On remarque que les fonds sont entrés sous forme de tableau Html qui est mis à jour régulièrement. Chaque ligne du tableau correspond à un fonds et pointe donc vers un fichier PDF à télécharger. Les fichiers doivent être stockés à l'emplacement suivant : Accueil/ Contribution/ Gamme financière/ Les nouveaux fonds. Dans ce répertoire, nous ferons très rarement des modifications, mais plus des ajouts. On va stocker les fichiers, récupérer leurs chemins d'accès, créer une ligne dans le tableau avec les éléments correspondant au fonds qu'on souhaite ajouter et enfin ajouter en lien le chemin d'accès ce fichier. Le nouveau fonds a ainsi été créé.

Très souvent comme nous l'avons dit, l'ajout d'un fonds entraîne le retrait d'un autre du tableau des nouveaux fonds. Les fonds qui sont retirés du tableau des nouveaux fonds sont mis dans le tableau des anciens fonds (voir image ci-dessous).



21/01/2022

Date de mise à jour : 10/01/2022

Retrouvez les fonds des sociétés de gestion partenaires du mois précédent.

Les fonds du mois précédent
Retrouvez tous les fonds des mois précédents

Société de gestion	Nom du fonds	Code ISIN
JP Morgan AM	JPM America Equity D (acc) - EUR	Fiche LU0217390490
CPR Asset Management	CPR Invest - Hydrogen - A EUR - Acc	Fiche LU2389405080
Colville Capital Partners France	Silver Autonomie R	Fiche FR00140052N5
La Financière de l'Echiquier	Echiquier Luxury A	Fiche FR0010960930

Figure 10: les fonds des mois précédents.

Pour récapituler, la gamme financière possède à elle seule plusieurs autres sous-rubriques sur lesquelles j'ai contribué en dehors de celles sur-citées. Je pense notamment aux partenaires gold et partenaire privilégiés ; les opérations sur les titres ; les gestions pilotées et les gestions

horizon ; les fonds euros et euro croissance ; les supports structurés et buy & hold, ces 3 derniers contenus dans la vie des supports.

C. Gestion du site B2B et B2C

La gestion de ces deux sites va se résumer à la publication des articles, la modification des bannières, la mise à jour des contenus ou tout simplement la création de nouvelles pages web. Les différentes contributions sur ces sites doivent être reportées dans l'outil Monday prévu à cet effet.

À la date d'aujourd'hui, je dénombre 15 demandes que j'ai traitées sur le site B2C sur une durée plus ou moins longue et 96 demandes que j'ai traitées sur le site B2B.

Ci-dessous, je vous montre certaines demandes que j'ai traitées sur les sites web de Cardif :



Les étapes clés de votre projet immobilier

Vous souhaitez réaliser votre rêve et devenir propriétaire ? Découvrez les 4 étapes clés de votre projet immobilier

Publié le 25 avril 2022

SOMMAIRE

1. [Définissez votre projet d'achat](#)
2. [Démarrez votre recherche de "bien"](#)
3. [Financez votre achat](#)
4. [Rendez-vous chez le notaire pour la signature de l'acte authentique](#)

1 Définissez votre projet d'achat

Commencez par identifier l'environnement de votre futur bien, comme la proximité : des écoles, des commerces, des transports, ou encore des espaces verts... Vous avez trouvé le quartier qui vous correspond ?

Définissez vos critères prioritaires pour choisir votre futur logement.

Étudiez ensuite les prix du marché local.

Définissez votre apport personnel et déterminez votre capacité d'emprunt avec l'aide de votre conseiller bancaire.

2 Démarrez votre recherche de "bien"

Plusieurs possibilités : Effectuez vos premières recherches en consultant les annonces immobilières ou avec l'aide d'un agent immobilier, ou d'un chasseur d'appartement.

Inspectez en détail le logement et l'immeuble s'il s'agit d'une copropriété.

Economisez plus de 10 000 € sur votre assurance de prêt

[Simuler mon tarif](#)

Pour votre devis d'assurance de prêt, contactez-nous au

0 800 05 56 56 Service & appel gratuits

Du lundi au vendredi de 9h à 18h et le samedi de 10h à 14h.

C'est simple et sans engagement !

Figure 11: exemple d'article publié sur le site cardif.fr



#RDVExperts : les têtes de série du mois de juin



Les stars ne se croisent pas uniquement autour du court Central ! Découvrez le nouveau programme de votre rendez-vous hebdomadaire

Chaque jeudi, retrouvez nos experts et invités pour partager des contenus exclusifs, avec au programme ce mois-ci des thématiques au cœur de l'actualité : assurance emprunteur, marchés financiers ou encore ingénierie patrimoniale.

Ci-dessous le détail de ces rendez-vous, à enregistrer dans votre calendrier :

- Jeudi 2 juin - #Podcast : **La loi Lemoine** par **Abdoulaye Samb Diop**, Responsable du service juridique Protection chez BNP Paribas Cardif
- Jeudi 9 juin - #Podcast : **Marchés européens : quelles perspectives ?** (partie 1/2) par **Nathalie Benatia**, Chef Macroéconomiste chez BNP Paribas Asset Management
- Jeudi 16 juin - #Podcast : **Marchés européens : quelles perspectives ?** (partie 2/2) par **Nathalie Benatia**, Chef Macroéconomiste chez BNP Paribas Asset Management
- Jeudi 23 juin - #Document : **Flash actualité patrimoniale** par le Service ingénierie patrimoniale et support juridique de BNP Paribas Cardif France.
- Jeudi 30 juin - #Podcast : **Inflation : quel(s) impact(s) sur vos placements ?** par **Mathieu Mucherie**, Chef Economiste, et **Serge Werlé**, Responsable de l'ingénierie financière au sein de BNP Paribas Cardif France.

Rendez-vous chaque jeudi dans vos boîtes mail ou sur [LinkedIn](#) pour les #RDVExperts ! Vous pouvez également retrouver l'ensemble de nos podcasts vidéo sur notre playlist [YouTube](#) dédiée et nos podcasts audio sur [Ausha](#).

Jeu, set et match !

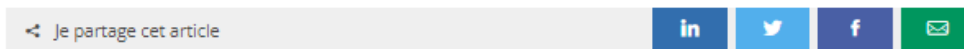


Figure 12 : exemple d'actualité publiée sur le site des partenaires

The screenshot shows the CARDIF website interface. At the top, there's a navigation bar with 'PARTICULIER' and 'PARTENAIRE' tabs, and a sub-header 'Site à destination des conseillers en gestion de patrimoine et aux courtiers en assurance.' Below this is the CARDIF logo and the tagline 'L'assureur d'un monde qui change'. The main navigation bar includes 'ACTUALITÉS', 'QUI SOMMES NOUS ?', 'NOS SOLUTIONS', 'PAROLES DE PARTENAIRES', 'CONTACT', and 'FINAGORA'. The central banner features a video player with a man in a suit and the text 'VOUS ACCOMPAGNER RESTE NOTRE PRIORITÉ' with the hashtag #AVOSCOTES. Below the banner are several article teasers: '12 DISTINCTIONS EN 2022' for 'EPARGNE RETRAITE PREVOYANCE', 'LES RENDEZ-VOUS EXPERTS UN SUJET, UN EXPERT JUILLET 2022', 'Cardif Elite Lux' (Le contrat d'assurance vie), and 'Cardif Libertés Emprunteur'.

Figure 13 : mise à jour d'une bannière sur le site des partenaires

Les produits
Epargne Cardif



Avant

Les produits
Epargne Cardif

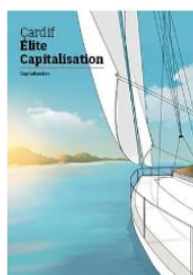
Après



Cardif Elite

Je compose mon contrat d'assurance vie sur-mesure, à partir de 15 000 €

[Découvrir notre offre](#)



Cardif Elite Capitalisation

Transmettez votre patrimoine en toute sérénité grâce à un contrat de capitalisation

[Découvrir notre offre](#)

Figure 14 : modification d'une LP dans le but de la rendre plus accessible et pour optimiser de l'espace

Information réglementaire

Conformément au règlement européen PRIIPS :

[Accédez aux documents d'information clés des contrats d'assurance BNP Paribas Cardif](#)

Conformément aux exigences de la loi PACTE :

[Consultez les taux moyens nets servis du fonds en euros \(contrats commercialisés par Cardif Assurance vie\)](#)

Conformément aux exigences de l'accord sur la transparence des frais :

[Consultez les tableaux de synthèse des frais des contrats d'assurance BNP Paribas Cardif](#)

Figure 15 : mise à jour d'une LP suite à l'ajout de nouveaux produits.



D. Gestion des différents projets

1. Projet d'accessibilité numérique

D'après l'article 47 de la loi n°2005-102 du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées, article que je paraphrase en disant : tous les services numériques publics doivent être accessibles à tous et surtout aux personnes handicapées. Ainsi donc, Les recommandations internationales pour l'accessibilité de l'internet doivent être appliquées pour les services de communication publique en ligne, quel que soit le contenu, le mode d'accès ou de consultation. Les règles relatives à l'accessibilité numérique sont fixées, les délais d'adaptation et les sanctions encourues par les organismes qui ne la respecteront pas sont aussi fixés. (Source : <https://www.legifrance.gouv.fr/loda/id/JORFTEXT000000809647/>).

Un site web ouvert au public va de ce fait être un site web permettant aux personnes handicapées d'accéder à son contenu sans avoir de difficulté. Pour dire d'un site qu'il est accessible, on peut vérifier les éléments suivants :

- Toutes les informations sont reportées lorsqu'on navigue à l'aide d'un outil de synthèse vocale. Si oui, on est certain qu'un mal voyant peut avoir accès à notre site facilement,
- La navigation au clavier permet d'atteindre toutes les informations pertinentes,
- Le site web reste correct, peu importe la taille de l'écran sur lequel on le consulte, peu importe l'agrandissement et l'espacement qu'on lui applique,
- Le contraste de couleur (couleur du texte / couleur de fond) doit être supérieur à 4.5,
- Chaque image porteuse d'information doit avoir un texte alternatif,
- Il n'existe pas des balises utilisées juste à des fins de décoration,
- Les pseudo-classes (hover, focus...) sont bien définies sur les éléments interactifs,

La liste reste non exhaustive. Lors de la création et/ou des modifications d'un site web, on doit tenir compte des éléments d'accessibilité de la norme en vigueur. Pour cela, il existe des référentiels comme le RGAA (Référentiel Général d'Amélioration de l'Accessibilité) pour mettre en conformité les applications numériques créées.

Afin de se conformer au décret d'application de la loi (2019) de l'article 47 de la loi de 2005, Cardif a décidé de faire un audit de ses applications web et de les rendre accessible au besoin. D'après le document d'information de Cardif que l'on peut consulter sur le lien 1 en annexe 1, il est marqué :

« BNP Paribas Cardif s'engage à rendre accessibles ses sites web (internet, intranet et extranet), ses applications mobiles, ses progiciels et son mobilier urbain numérique conformément à l'article 47 de la loi n°2005-102 du 11 février 2005.

[...] Nous présentons nos excuses à toutes celles et ceux qui ne pourraient pas, à ce jour, accéder de manière simple et complète à l'ensemble du contenu proposé. »

Ils informent également que l'audit du site a révélé un niveau faible d'accessibilité. C'est de là que vient le projet « d'accessibilité numérique » chez Cardif, projet sur lequel j'ai eu le plaisir de travailler avec l'équipe en charge.

NB : Cardif dispose d'un site de recette pour faire les tests avant publication et d'un site en production pour le code final à destination des internautes. Nous avons effectué toutes nos

modifications d'accessibilité sur le site en recette. Ces modifications ne seront portées en production qu'une fois que le site aura atteint un taux convenable d'accessibilité.

Afin de rendre accessible le site de Cardif, nous sommes intervenus à plusieurs niveaux :

a. Accessibilité sur les images

Elle reste la plus facile.

- Lorsque la balise image est porteuse d'information, une alternative textuelle doit être passée à cette balise au travers de l'attribut '**alt**'. L'alternative doit être la plus précise et la plus descriptive possible.
- Lorsque la balise est utilisée à une fin de décoration, l'attribut '**alt**' doit être vide (**alt= ""**). Cependant, les balises possédant un attribut **role="img"** ne peuvent faire office d'image de décoration qu'à la condition qu'elles possèdent un attribut **aria-hidden="true"**.
- L'utilisation des images contenant des textes est déconseillée, nous avons préféré reprendre le texte présent sur ces images en CSS ou de les remettre intégralement dans l'attribut '**alt**'.
- Lorsque nous avons un groupe d'images, nous les regroupons en une seule image et donnons une alternative textuelle qui décrit l'image.

b. Accessibilité sur les Listes

La notion d'accessibilité pousse à privilégier l'utilisation des listes (ordonnées ou non ordonnées) pour décrire des contenus proches et argumentant la même thématique. Nous nous devons donc de regrouper visuellement sous forme de liste nos contenus web. La notion de 'regroupement visuel' fait référence à :

- La présence de marqueur de liste pour permettre d'identifier visuellement qu'il s'agit d'une liste non ordonnée (puces, points, images, icônes...),
- La présence de marqueur tel qu'un numéro ou une lettre pour faire comprendre qu'il s'agit d'une liste ordonnée (lettres ou chiffres suivant l'ordre).

c. Accessibilité sur les liens

Pour rendre notre site web accessible sur le critère de lien, nous avons dû intervenir à différents niveaux :

La navigation au clavier est un élément important de l'accessibilité. Les liens constituent le fondement du web et un élément incontournable pour un site web, il est nécessaire et primordial qu'en naviguant au clavier qu'on puisse atteindre les liens présents sur le site. Les liens doivent avoir le focus lorsqu'on navigue au clavier et être reconnaissable de façon distincte au survol avec la souris.

- **Le focus** : pour parfaire le focus sur les liens du site, nous avons désactivé le soulignement par défaut. À la navigation au clavier, on active la bordure autour de l'élément.
- **Le hover** : au survol d'un lien, on va faire apparaître le soulignement. On identifie donc plus facilement le lien qui est interactif contrairement aux autres éléments du contenu web.

d. Accessibilité sur les balises de décoration

Sur le site en production aujourd'hui, il existe plusieurs balises de décoration. C'est le cas de :

- La balise '**div**', '**span**' et/ou '**i**'. Cette balise est souvent utilisée pour ajouter des icônes à l'aide des classes Css. Comme on peut le voir sur la figure ci-dessous, en 2 on peut voir la balise qui sert à obtenir l'icône 1. Cependant, cette balise ne vérifie pas les critères d'accessibilité, car elle est vide. Un lecteur d'écran qui parcourt cette balise ne restituera rien. Pour résoudre ce souci, nous devons dire au lecteur d'écran de ne pas prendre en compte cette balise. Ceci se fait à l'aide de l'attribut '**aria-hidden**' qui prend la valeur '**true**'.



Figure 16 : accessibilité sur la balise `` de décoration.

- La balise `<hr>`. Sur le site de la production, la balise `<hr>` est souvent utilisé pour faire office de soulignement comme on peut voir sur le 1 et 2 de la figure ci-dessous. Son avantage est que l'on peut définir la taille du trait. L'équipe UI/UX avoir le trait à un niveau bien précis du texte, donc impossible d'utiliser un soulignement classique. Cependant, pour des raisons d'accessibilité, nous ne pouvons pas utiliser une balise `<hr>` car d'après la documentation Html, elle doit être utilisée pour changer de thématique (voir lien 2 en annexe 1).

Pour résoudre ces problèmes, nous avons opté pour l'ajout d'une bordure à partir du pseudo élément '**before**' et l'ajout de certaines classes comme on peut le voir en 3 sur la figure ci-dessous.

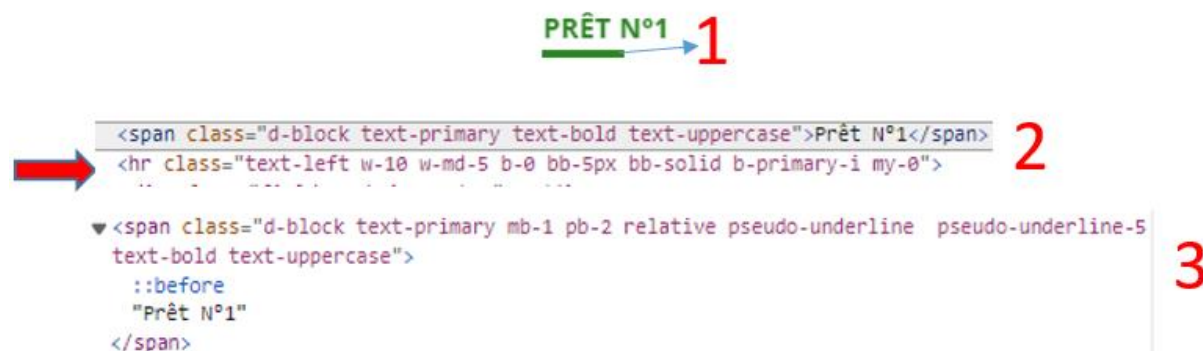


Figure 17: accessibilité sur la balise <hr> de décoration.

En 3, nous ajoutons :

- La classe '**relative**' pour donner la position relative à l'élément qui est le dans ce cas.
- La classe '**pseudo-underline**' qui s'applique sur le pseudo-élément *before* va créer un nouveau contenu vide, le placer en position absolu par rapport au premier parent qui est en position relative (notre span), placer ce contenu en bas à droite, lui appliquer une bordure de 'Solid' en bas uniquement avec une épaisseur de 5 px, et lui donner une taille par défaut de 13% de la longueur de l'élément **span**.
- La classe '**pseudo underline-5**' va permettre de modifier à 5% la taille qui a été fixé à 13% précédemment.

```

1  #root .relative {
2    position: relative !important;
3  }
4
5  #root .pseudo-underline::before {
6    content: "";
7    position: absolute;
8    left: 0;
9    bottom: 0;
10   border-bottom: solid;
11   border-bottom-color: #0d7d50;
12   border-bottom-width: 5px;
13   width: 13%;
14 }
15
16 #root .pseudo-underline-5::before {
17   width: 5% !important;
18 }

```

Figure 18 : classes nécessaires pour corriger l'accessibilité des balises 'hr'

Ayant remarqué que c'est un problème qui revient souvent et dont la taille de la bordure n'est pas toujours pareille, nous avons décidé d'inclure dans le Css du site web un ensemble de classe qui feront varier cette taille. Nous avons aussi tenu compte du fait que la bordure puisse être à gauche et pas à droite, en haut ou en bas.

```

/***** Fix hr border *****/
#root .pseudo-underline::before {
  content: "";
  position: absolute;
  bottom: 0;
  border-bottom: solid;
  border-bottom-color: #0d7d50;
  border-bottom-width: 5px;
  width: 13%;
}

#root .pseudo-underline-left-5::before {
  left: 0;
  width: 5% !important;
}

#root .pseudo-underline-left-10::before {
  left: 0;
  width: 10% !important;
}

#root .pseudo-underline-left-15::before {
  left: 0;
  width: 15% !important;
}

#root .pseudo-underline-right-20::before {
  right: 0;
  width: 20% !important;
}

#root .pseudo-underline-left-20::before {
  left: 0;
  width: 20% !important;
}

#root .pseudo-underline-right-5::before {
  right: 0;
  width: 5% !important;
}

#root .pseudo-underline-right-10::before {
  right: 0;
  width: 10% !important;
}

#root .pseudo-underline-right-15::before {
  right: 0;
  width: 15% !important;
}

```

Figure 19 : ensemble de classes pour rendre accessible le soulignement

e. Accessibilité sur les formulaires

Afin de rendre un contenu de type formulaire accessible, plusieurs éléments sont à prendre en compte.

- Utilisation de la balise `<label>` et des attributs **for** & **id** :
 - Lorsque l'étiquette d'un champ doit être visible, il faut utiliser la balise `<label>` pour l'étiqueter.
 - Utiliser l'attribut **for** dans les labels et **id** dans les champs.
 - Chaque couple champ/label doit contenir la même valeur pour l'attribut **for** et **id**. De plus, ceux-ci doivent être uniques (voir image ci-dessous).

```
1 <label for="nom">Votre nom</label>
2 <input type="text" id="nom" name="nom" autocomplete="family-name" />
```

Figure 20 : accessibilité label/champ d'un formulaire

- Utilisation de l'attribut **title** : elle va être utile pour décrire des champs qui n'ont pas une étiquette. Il doit être renseigné en décrivant la fonction du champ. Cet attribut va trouver toute son importance lorsque le site est parcouru par un lecteur d'écran, la valeur de cet attribut sera restituée. L'attribut **placeholder** ne peut remplacer **title**.
- Utilisation de l'attribut **autocomplete**. Comme nous pouvons le voir sur la figure ci-dessus, j'ai ajouté cet attribut sur tous les champs de formulaire. Sa valeur est fonction du champ. L'utilisation de cet attribut facilite le remplissage des champs qui constitue des informations personnelles.
- Utilisation des balises de regroupement tel que `<fieldset>`. Comme nous pouvons le voir ci-dessous, elle est utilisée pour regrouper des groupes de champ dont l'intitulé est similaire. Nous nous sommes assurés au fur et à mesure que tout le formulaire est accessible à la navigation au clavier.

```
1 <fieldset>
2   <legend class="h3 d-block text-90 text-bold mb-3">Civilité
3   </legend> <label class="mb-0" for="mme-mrh-civilite-field"> <input checked="checked" id="mme-mrh-civilite-field"
4     name="user[civility]" type="radio" value="MME"> <span
5     class="checked-bg-primary checked-b-primary text-90">Madame</span>
6   </label> <label class="mb-0" for="m-mrh-civilite-field"> <input id="m-mrh-civilite-field" name="user[civility]"
7     type="radio" value="M"> <span class="checked-bg-primary checked-b-primary">Monsieur</span>
8   </label>
9 </fieldset>
```

Figure 21 : accessibilité sur un formulaire avec la balise `fieldset`

f. Accessibilité sur les accordéons

La version des accordéons en production aujourd'hui ne permet pas de naviguer au clavier. L'objectif a donc été de rendre cette version navigable au clavier et que l'élément qui possède le curseur soit identifiable de façon distincte.

Sachant que Liferay 7 utilise la version 4 de Bootstrap, nous avons jugé plus simple de passer par ce Framework d'interface utilisateur qui offre une bibliothèque de classe importante pour résoudre ce problème.

Pour faire simple, afin de rendre un accordéon accessible, certaines classes et attributs doivent être présents.

- Sur le bouton qui fait l'action :
 - ***Data-toggle = "collapse"*** c'est cet attribut qui fait l'action **hide** / **show** à l'aide de JavaScript.
 - ***Data-target = "#collapseOne" & aria-control = 'collapseOne'*** : ces attributs désignent vers l'élément qui va subir l'action.
 - Le dernier et le plus important est ***aria-expanded="true"***. C'est lui qui indique aux lecteurs d'écran si l'élément est déroulé ou non.
- Sur la ***div*** qui subit l'action Collapse :
 - L'attribut ***id*** est nécessaire.
 - Un attribut '***data-parent***' et un autre '***aria-labelledby***' ne sont pas obligatoires, mais ils viendraient parfaire l'accessibilité.
 - En fin la classe collapse qui va servir à faire un ***display = 'none'*** sur le contenu (masque le contenu).

Consulter l'annexe 3 pour voir le résultat des accordéons que j'ai rendu accessible. De manière analogue, j'ai rendu accessible tous les accordéons du site web.

2. Projet d'optimisation du temps de formatage

Afin de faciliter la tâche aux développeurs de la Digital Factory qui font de la publication d'article, j'ai décidé de créer une page web pour formater les articles en Html.

a- Contexte et problématique

Deux mois après avoir rejoint l'équipe, j'étais déjà opérationnel et autonome. J'ai pendant ces premiers mois appris et je me suis familiarisé avec les outils qu'utilise mon équipe au quotidien. J'ai cependant remarqué que la publication d'un article demandait beaucoup de temps. Ce temps était long à cause d'une étape clé qui est la plus fastidieuse. Je me suis donc posé la question de savoir s'il est possible de palier à ce problème. Ne souhaitant plus passer autant de temps pour formater juste un article, j'ai décidé de me pencher sur la question et d'en faire un problème auquel je dois trouver la solution.

b- Justification des choix faits

À cette période, je m'auto formais sur la bibliothèque JavaScript qu'est React JS. Il n'a pas fallu chercher longtemps pour que je me dise : « c'est cette bibliothèque que je veux pour résoudre ce problème ». Un élément principal m'a orienté à l'aveugle sur cette solution :

- Le JSX. Cette syntaxe particulière qui est en fait une extension de JavaScript qui utilise le Html pour décrire à quoi va ressembler l'interface utilisateur.
- La place de React dans le développement web des IU aujourd'hui. C'était pour moi le timing parfait pour mettre en pratique les bases que je venais d'apprendre et d'en apprendre davantage
- Enfin tout simplement le fait que React est la bibliothèque par excellence pour ce projet particulier.

Après ma décision de résoudre ce problème et après avoir trouvé le langage à utiliser, j'en ai discuté avec mon manager en lui présentant l'idée et les plus que la solution peut apporter à l'équipe. Il a tout de suite été séduit par l'idée et m'a donné son feu vert.

Après son accord qui a suivi les réponses aux questions « pour qui, pourquoi et avec quoi développer cette page web ? », j'ai commencé à réfléchir sur les fonctionnalités que j'attends de mon application ; et comment je peux les mettre en œuvre.

c- Description des fonctionnalités

1. En tant qu'utilisateur de la page web, je dois pouvoir :
2. Entrer l'intitulé d'un sous-titre, son numéro et récupérer en sortie du code Html qui correspond à la syntaxe de Cardif.
3. Entrer un texte et obtenir en sortie un ou plusieurs paragraphes en fonction du nombre de retours à la ligne du texte.
4. Entrer des éléments d'une liste avec des retours à la ligne et obtenir en sortie une liste formatée en Html et toujours suivant la syntaxe de Cardif.
5. Entrer l'url d'une vidéo, faire un retour à la ligne, entrer le titre de la vidéo et récupérer en sortie un « iframe » correspondant au format iframe de Cardif.

6. Disposer d'un box d'éléments contenant tous les éléments (sous-titre, liste, paragraphe, vidéo et box 'bon à savoir').
7. Ajouter des éléments () que je souhaite utiliser pour formater mon article.
8. Supprimer un composant quand je n'en ai plus besoin,
9. Formater un seul élément et obtenir le code correspondant.
10. Formater tout un article et obtenir le code correspondant.
11. Modifier un élément incorrect de façon instantanée.
12. Disposer d'un élément me permettant de valider mon formatage (bouton, lien).
13. Être automatique redirigé vers le contenu important lorsque je valide le formatage

d- Description du projet

Comme nous l'avons mentionné plus haut, le projet a été réalisé avec la bibliothèque JavaScript qu'est React JS.

Il faut retenir deux choses de React :

- Il est déclaratif : on définit des vues simples pour chaque état de l'interface utilisateur. Une fois qu'un changement doit être fait, React s'en occupe de façon optimale en ne modifiant que le contenu qui a changé d'état sans toucher ceux qui n'ont pas changé.
- Se structure à base de composant : on crée des composants autonomes qui vont constituer l'application et ainsi favoriser le débogage de celle-ci.
- React fonctionne avec un DOM virtuel : une représentation du DOM est stockée en mémoire et React se charge de la réconciliation.

Je vais vous expliquer ici les différentes étapes qui ont permis d'obtenir notre page web.

- **Création des composants :** j'ai commencé par la création des composants qui vont constituer mon application. Il s'agit ici des composants liste, paragraphe, sous-titre, vidéo et box « bon à savoir » comme nous pouvons voir sur l'image ci-dessous qui montre le composant liste.

```

1 import React from "react";
2 import PartTitle from "../PartTitle";
3
4 class Liste extends React.Component {
5   constructor(props) {
6     super(props);
7     this.state = { key: this.props.givenKey }
8   }
9   render() {
10    return (
11      <div className="p-2 d-grid gap-2">
12        <PartTitle titre = "Liste"></PartTitle>
13        <textarea className="form-control" ref={this.props.reference}
14          id="liste" style={{ height: '120px' }} placeholder="Entre les elements de la liste" />
15        <button type="button" className="col-6 ms-auto btn btn-danger"
16          onClick={() => this.props.onDeleteEl(this.state.key)}> Supprimer</button>
17      </div>
18    );
19  }
20 }
21 export default Liste;

```

Code

Liste

Liste 1

Liste 2

Liste 3

Résultat

Supprimer

Figure 22 : création du composant liste

Dans notre application, il existe des composants qui ont été créés pour factoriser le code. Il n'est pas nécessaire de le présenter. C'est le cas d'un composant créé pour ajouter un titre à un autre composant, ce composant de titre est réutilisé dans tous nos éléments, liste, paragraphe...

Dans le code, on peut voir que la classe Liste hérite de la classe React Component. Son constructeur définit un état de l'application qui peut varier selon la valeur de la clé '*key*'. La classe liste va retourner du JSX. Il s'agit d'une div qui va contenir :

- **Un titre.** Ce titre est passé au composant Liste grâce au composant PartTitle qui prend en attribut titre='Liste'.
- **Un textarea.** Il a un attribut *ref* qui lui sera donné par son parent. Nous reviendrons sur l'utilité des références plus tard
- **Un bouton 'supprimer'.** C'est grâce à ce bouton qu'on va pouvoir supprimer un élément si l'on n'en a plus besoin (voir fonctionnalité 8 dans la section II, D, 3, c). La gestion de l'évènement est faite à partir d'une méthode de la classe parente et on lui passe en paramètre la clé du composant courant.

➤ **Création du box de boutons :** il s'agit ici du box qui va contenir tous nos boutons représentant les différents éléments. Il est lui aussi un composant au sens React. Il prend en paramètre les propriétés de son ancêtre le plus proche. Il utilise la méthode *choix_btn* () de la classe ancêtre. Cette étape s'appelle la remonter d'état, à la seule différence qu'au lieu de faire remonter un état localement défini, on fait remonter des constantes en fonction du choix de l'utilisateur.

```

1 import React from 'react';
2
3 const BoxDeComposant = (props) => {
4
5
6   return (
7     <div className=""
8       <div className=" px-1">
9         <button type="button" className="text-bold col-12 btn btn-outline-primary" onClick={() => props.choix_btn(0)}>Sous Titre </button>
10       </div>
11       <div className="row pt-2 p-0 m-0 ">
12         <div className="col-md-6 col-12 p-1">
13           <button type="button" className="text-bold col-12 btn btn-outline-primary" onClick={() => props.choix_btn(1)}> Paragraphe </button>
14         </div>
15         <div className="col-md-6 col-12 p-1">
16           <button type="button" className="text-bold col-12 btn btn-outline-primary" onClick={() => props.choix_btn(2)}> Liste</button>
17         </div>
18       </div>
19       <div className="row pt-2 p-0 m-0 ">
20         <div className="col-md-6 col-12 p-1">
21           <button type="button" className=" text-bold col-12 btn btn-outline-primary" onClick={() => props.choix_btn(3)}> >video </button>
22         </div>
23         <div className="col-md-6 col-12 p-1">
24           <button type="button" className=" text-bold col-12 btn btn-outline-primary" onClick={() => props.choix_btn(4)}> Box Bon a savoir </button>
25         </div>
26         <div className=" px-1 pt-5">
27           <a type="button" className="text-bold col-12 btn btn-success" href="#floatingTextareaz" onClick={() => props.choix_btn(5)}> Valider le Formulaire </a>
28         </div>
29       </div>
30     </div>
31   );
32 };
33
34 export default BoxDeComposant;

```

Figure 23 : code du box d'éléments

Pour résumer, nous avons créé nos éléments (composants), on a créé le box de bouton pour afficher un élément en fonction du clic sur un bouton du box (voir image ci-dessous). Nous allons à présent voir ce qui se passe lorsqu'on clique sur un bouton de ce box, donc ce que fait la méthode *choix_btn* ().

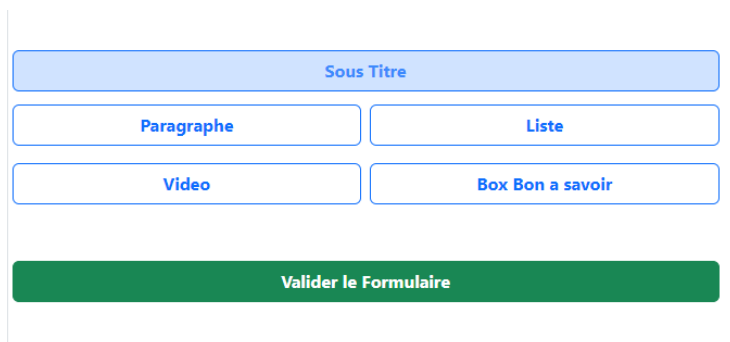


Figure 24: résultat du box de bouton

➤ **L'application dans le fichier App.js.** C'est la partie de notre application la plus importante.

Le code sur l'image ci-dessous montre le rendu de la classe App qui sera fait au DOM React. Il est scindé de la façon suivante :

- Header à la ligne 165,
- La div qui va contenir les entrées de l'application entre la ligne 168 et 173,
- La div qui va contenir la sortie de l'application entre la ligne 175 et 185,

Seule la div des entrées va retenir notre attention.

```
162 render() {
163   return (
164     <div className="">
165       <Header> </Header>
166       <div className="container p-2 border">
167         <div className="row">
168           <div className="col-6 ps-1 pe-0">
169             {this.FromMapToArray(this.Map_All_Elt).map((e) => e)}
170             <br></br>
171
172             <BoxDeComposant choix_btn={this.choix_btn} />
173           </div>
174
175           <div className="col-6">
176             <div className="form-floating">
177               <textarea
178                 className="form-control"
179                 readOnly
180                 defaultValue={this.state.valueToScreen}
181                 id="floatingTextarea2"
182                 style={{ height: "1000px" }}
183               />
184             </div>
185           </div>
186         </div>
187       </div>
188     </div>
189   );
190 }
191 }
```

Figure 25: rendu de app.js

Cette div a une instruction JS et une balise 'composant' :

i. Balise composant : Box de composant

On va noter ici la valeur de l'attribut choix_btn qui est une méthode de la classe App. Pour mieux comprendre le fonctionnement de cette fonction choix_btn, regardons cet extrait de la classe App. On voit que la classe App définit :

- **Map_ALL_Elt** (ligne 12). Lorsqu'on clique sur un bouton pour afficher un élément graphiquement, cet élément va d'abord être stocké dans cette Map.
- **Ref : qui est aussi une Map** (ligne 21). Lorsqu'on clique sur un bouton du box de bouton dans le but d'ajouter un élément, on doit créer une référence cet élément, cette référence est stockée dans cette Map. Nous l'utiliserons pour garder une traçabilité sur tous les éléments créés dans l'application. Notamment lors de la suppression d'un élément.
- La variable d'État **increment_nbre_elts**. Elle joue un rôle crucial dans notre application, elle s'incrémente quand on clique sur un bouton pour ajouter un élément

NB : c'est le changement d'état de cette variable qui entraine une mise à jour de l'état du DOM virtuel et donc ce qui aura changé dans notre classe App sera automatiquement mis à jour dans l'interface utilisateur.

```

11 class App extends React.Component {
12   Map_All_Elt = new Map(); // map qui contient tous mes Composant
13
14   constructor(props) {
15     super(props);
16     this.state = {
17       valueToScreen: '',
18       increment_nbre_elts: 0 // var qui va etre incrementé quand on ajoute un Composant
19     };
20
21     this.Ref = new Map() //Creation d'un tableau de reference qui va correspondre a chaque composant
22     this.choix_btn = this.choix_btn.bind(this);
23     this.deleteElt = this.deleteElt.bind(this);
24   }

```

Figure 26: constructeur de la classe App

Pour revenir à notre fonction choix_btn () :

- Elle est appelée lorsqu'on clique sur un bouton du box (sous-titre, paragraphe, liste, vidéo...).
- Chaque bouton de ce box appelle la fonction avec un paramètre qui lui est propre ; valeur 0 pour un sous-titre, valeur 1 pour un paragraphe, valeur 2 pour une liste, valeur 3 pour une vidéo, valeur 4 pour un pour le box « bon à savoir » et la valeur 5 pour le bouton de validation du formulaire.
- Si la valeur reçue est différente de 5, 0 par exemple, on a le code suivant pour ajouter un sous-titre.

```

95 //ajout sous titre
96 if (choix === 0) {
97   //cree une ref pour l'object suivre, et on l'ajoute dans le tableau de reference
98   this.Ref.set(this.state.increment_nbre_elts, [React.createRef(), React.createRef()])
99
100   //on passe la ref au composant et on stocke ce composant dans la variable elt_to_add
101   let elt_to_add = <SousTitre onDeleteElt={this.deleteElt} givenKey={this.state.increment_nbre_elts}
102     reference1={this.Ref.get(this.state.increment_nbre_elts)[0]} key={this.state.increment_nbre_elts}
103     getNumTitle={this.reference_du_numero_de_titre} reference2={this.Ref.get(this.state.increment_nbre_elts)[1]} />
104
105   //ajout de le composant au map cle = l'etat de la variable d'incrementation valeur => composant crée
106   this.Map_All_Elt.set(this.state.increment_nbre_elts, elt_to_add)
107 }

```

Figure 27: code exécuté dans la fonction choix_btn pour le clic sur sous-titre

À la ligne 98, on crée une référence et on la stocke dans la Map des références. Dans ce cas précis, on crée deux références pour ce composant, car l'une est pour le titre et l'autre pour le numéro du titre. Dans ce cas, à cette clé dans la Map, on va attribuer un tableau de 2 valeurs (2 réf.).

À la ligne 101, on crée un objet sous-titre (c'est l'initialisation de l'objet sous-titre qu'on a créé précédemment) et on lui attribue la référence créée à la ligne 98. Pour l'instant cet objet va être stocké dans la Map prévu pour contenir les différents composants. À la sortie de la fonction choix_btn, on incrémente la variable **increment_nbre_elts**. Le changement d'état de cette variable va mettre à jour le DOM virtuel et donc notre composant va apparaître dans la liste des composants sur l'interface utilisateur.

- Si la valeur reçue est 5, l'utilisateur souhaite valider le formulaire et obtenir son résultat.

Pour chaque pair key/value de la map Ref, on vérifie que la valeur de la Ref existe et n'est pas nulle. En fonction de l'id de l'élément (paragraphe, vidéo...), on applique une fonction de formatage particulière (voir image ci-dessous) parmi les fonctions de formatage qu'on verra plus tard. Le texte formaté est incrémenté à la variable valeur (voir ligne 77) qui sera retourné. Une fois la boucle terminée, on modifie l'état de la variable valueToScreen (voir ligne 82)

présente dans le constructeur. Une fois de plus, c'est la modification d'une variable d'état qui va mettre à jour l'UI.

```
60 // choix 5 pour valider le formulaire
61 if (choix === 5) {
62   let valeur = ''
63   for (let i of this.Ref) {
64     if (i[1].length !== 2 && i[1].current.value && i[1].current.value !== '') {
65       if (i[1].current.id === 'paragraphe') { valeur += format_paragraphe(i[1].current.value) }
66       if (i[1].current.id === 'liste') { valeur += format_liste(i[1].current.value) }
67       if (i[1].current.id === 'video') { valeur += format_video(i[1].current.value) }
68       if (i[1].current.id === 'asavoir') { valeur += Format_Box_A_Savoir(i[1].current.value) }
69     }
70     if (i[1].length === 2 && i[1][0].current.value !== '') {
71       // certains composants necessite de leur passer deux ref donc on y accede comme dans un tableau a 2 dimensions
72       if (i[1][0].current.id === 'sous titre') {
73         valeur += format_sous_titre(i[1][0].current.value, i[1][1].current.value)
74       }
75     }
76   }
77   this.setState({
78     valueToScreen: valeur
79   })
80 }
81 }
82 }
83 }
84 }
85 }
```

Figure 28 : fonction `choix_btn` pour un choix de valider le formulaire

Maintenant que les attributs et méthodes de la classe App ont été expliqués, il sera plus facile d'expliquer le fonctionnement de la méthode. Supprimer un élément de l'affichage va revenir à retirer sa référence dans la table des références et de le retirer lui aussi de la table des éléments. Ainsi, lors du prochain changement d'état, cet élément ne sera plus présent dans les boucles.

NB : Sans consulter le code, un expert React peut me poser au premier abord la question de savoir pourquoi le choix des composants non contrôlé alors que React le recommande ? pendant le développement avec les composants contrôlés, j'ai eu du mal à garder la traçabilité sur les éléments qui sont ajoutés de façon aléatoire. Je dois savoir exactement quel composant est ajouté, quel est modifié, et quel est supprimé ; aussi et surtout pouvoir supprimer un composant particulier. Il est parfois fastidieux d'utiliser les composants contrôlés, car il vous faut écrire un gestionnaire d'événement pour chaque possibilité de changement des données, et gérer toute modification des saisies via un composant React. Donc dans mon cas, j'ai dû passer par les composants non contrôlés pour avoir la main sur chaque composant qui est créé afin d'y apporter mes modifications et mes suppressions. C'est là la nécessité de donner à chaque composant une référence. Pour en savoir plus, voir l'annexe 1, lien 3 et 4

ii. Instruction JS

Il s'agit de l'instruction `{this.FromMapToArray(this.Map_All_Elt).map((e) => e)}`. Elle permet simplement d'afficher sur l'interface utilisateur les composants présents dans la collection de composants **Map_All_Elt**.

Pour récapituler :

- J'ai créé des composants et un box de bouton correspondant chacun à un composant.
- Chaque composant est contrôlé à l'aide d'une fonction `choix_btn` qui prend un paramètre de 0 à 5.
- Si ce paramètre est différent de 5, on crée une référence React, on crée un composant non contrôlé auquel on attribue cette référence. On stocke cet élément dans une

collection de composant. On modifie l'état de l'application pour mettre à jour et afficher le composant qui a été créé.

- Si ce paramètre est 5, les valeurs des inputs dont on dispose de la référence sont formatées à l'aide des fonctions adéquates. Ces valeurs sont jointes et passé à la variable d'état qui va permettre de mettre à jour le DOM React et donc d'afficher le résultat attendu par l'utilisateur.
- À la suppression d'un composant, on valide à nouveau le formulaire, ce qui permet de mettre à jour l'affichage.

e- Présentation des fonctions de formatage utilisées

Pour résoudre notre problème, nous avons dû utiliser certaines expressions rationnelles. Elles sont listées dans le tableau ci-dessous.

Tableau2 : des expressions rationnelles utiles pour les fonctions de formatage

Caractères	Significations
{n,}	Il y a au moins n occurrences. Exemple : /a{3,}/ = 'aaa', 'aaaa' mais pas 'aa'
\s	Correspond à un espace, une tabulation ou un saut de ligne.
\t	Tabulation
/[^A-Za-z0-9_]/	Correspond à tous les caractères non alphanumériques tels que : %, @, \$, £, "...
\n	Retour à la ligne
\g	Sélectionne toutes les occurrences
^	Exprime la négation d'une expression

- Formatage des paragraphes : je veux pour voir récupérer en entrant un texte et renvoyer un ou plusieurs paragraphes en fonction de la structure du texte. Le résultat doit être un code formaté en Html avec la balise '<p>'.

Le texte est découpé en paragraphe avec la méthode split () qui prend en paramètre une expression rationnelle qui est « \n » et stocké dans un tableau. Ensuite, pour chaque élément du tableau, on ajoute une balise de paragraphe ouvrante et une fermante.

```
// cette fonction récupère un texte et y ajoute des balises html de paragraphe necessaires
function format_paragraphe(params) {
  let table_paragraphe = params.split('\n') // transforme le texte en tableau de paragraphe
  let resultat_paragraphes = []
  table_paragraphe.forEach(element => {
    if (element.length !== 0) { // paragraphe vide ?
      resultat_paragraphes.push('<p>\n\t' + element + '\n</p>\n\n') // add <p> ... </p> à chaque paragraphe
    }
  });
  return resultat_paragraphes.join(' '); // joint tous les paragraphes
}
```

Figure 29 : fonction de formatage des paragraphes

- Formatage des sous-titres : l'idée est de récupérer un sous-titre et son numéro afin de renvoyer le code Html qui correspond à la syntaxe de code des sous-titres au sein de la Digital Factory (DF). Au sein de la DF, dans un article, tout sous-titre doit avoir un

identifiant. Pendant le formatage des sous-titres, il faut ajouter cet identifiant unique en attribut.

L'identifiant correspond au titre auquel on retire les accents, on remplace les espaces par les tirets et on retire tout caractère qui n'est pas alphanumérique. Pour une question d'esthétique et de prévoyance, on retire les espaces en double puis les espaces en début et en fin de phrase.

On peut voir en annexe à quoi ressemble une ensemble sous-titre + paragraphe sur le site de Cardif (Voir annexe 3) et comment se présente le code formaté avec la page web conçu (Voir annexe 3).

```
// formater un sous titre
function format_sous_titre(params, num) { // fonction qui prend en parametre un titre et un numero de titre
  // remplace les accents dans notre titre afin de constituer l'id du sous titre
  let id = params.replaceAll('é', 'e')
    .replaceAll('à', 'a')
    .replaceAll('â', 'a')
    .replaceAll('ô', 'o')
    .replace(/[\^A-Za-z0-9\s]/g, "").replace(/\s{2,}/g, " ").trim() // remplace les caracteres non alphanumerique par "",
    // " " = " ", retire les espaces au debut et de fin avec la methode trim() , //
    .replaceAll(" ", "-") // " " = "-"
    .toLowerCase(); // texte en minuscule
  let format_title = '<div class="d-flex">\n\t<span class="title-number h3 text-fuchsia z-depth-5">\n\t\t' + num + '\n\t</span>\n\t<h2
  + 'id="' + id + '">\n\t\t' + params + '\n\t</h2>\n</div>\n\n' // ajouts des balises autour des valeurs de variable

  return (
    format_title
  );
}
```

Figure 30 : fonction de formatage des sous-titres

- Formatage des listes : je veux ici constituer un élément de liste avec chaque paragraphe d'un texte. Je prends un texte en entrée, je le découpe en fonction des retours à la ligne, j'entoure ces paragraphes avec les balises ..., je joins ces paragraphes et je concatène le tout avec les balises ... toute fois avec les classes qu'il faut.

```
function format_liste(text) {
  let ul_debut = '<ul class="list-styled list-bullet-primary">';
  let ul_fin = '\n</ul>\n\n';
  let liste_elt_li = []
  text.split('\n').forEach(element => { //chaque paragraphe du texte constitue un element de liste
    if (element.length !== 0) {
      liste_elt_li.push('\n\t<li>' + element + '</li>') // si l'element est non nul, on l'entoure des balises <li> ... </li>
    }
  });
  let result_liste_html = ul_debut + liste_elt_li.join(' ') + ul_fin // ajout des <ul> autour de tous nos <li> precedents
  return result_liste_html;
}
```

Figure 31 : fonction de formatage des sous-titres

- Formatage du box « bon à savoir » : pour ce cas principalement, je veux appliquer un style particulier au premier paragraphe uniquement ; je le découpe donc en paragraphe comme précédemment. J'insère les éléments dans ce tableau de façon ordonnée. Puis, je joins ses éléments que j'envoie à la sortie de la fonction.

```
// le premier paragraphe du texte en parametre constitue le titre de du box
function Format_Box_A_Savoir(textValue) {
  let table_paragraphe = textValue.split('\n')
  let table_resultat = []
  table_resultat.push('<div class="bg-grey-10 p-5 mb-5" >\n<h3 class="text-primary">\n\t\t')
  table_resultat.push(table_paragraphe[0])
  table_resultat.push('\n\t</h3> \n\t<p>\n\t\t')
  for (let index = 1; index < table_paragraphe.length; index++) {
    table_resultat.push(table_paragraphe[index])
  }
  table_resultat.push('\n\t</p> \n</div>\n')
  return table_resultat.join(' ')
}
```

Figure 32 : fonction de formatage du box « bon à savoir »

- Formatage d'un iframe vidéo : l'input contient un texte avec 2 paragraphes. Le premier est le lien de la vidéo et le second est le titre. De façon analogue aux façons précédentes, le découpe le texte. Ces deux éléments vont être concaténés à des endroits précis de l'iframe. En sortie, je revois un iframe pouvant être utilisé.

```
function format_video(link_and_title) {
  let link = link_and_title.split("\n")[0]
  let video_title = link_and_title.split("\n")[1]

  let iframe = '<div class="video-container my-5"><iframe allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture" allowfullscreen="" frameborder="0" src="' + link + ' title="Video youtube: ' + video_title + '" style="width: 560; height: 315;"></iframe></div> '+
  '\n <style type="text/css">.video-container { \n \t position: relative; \n \t padding-bottom: 56.25%; '+
  '\n \t height: 0; \n \t overflow: hidden; \n \t } \n \t .video-container iframe, \n \t .video-container object, '+
  '\n \t .video-container embed { \n \t position: absolute; \n \t top: 0; \n \t left: 0; \n \t width: 100%; '+
  '\n \t height: 100%; \n \t } \n </style>'

  return iframe + '\n'
}
```

Figure 33 : fonction de formatage iframe vidéo

f- Présentation des résultats

J'ai travaillé avec le logiciel de gestion des versions GitHub pour conserver l'évolution du projet. Le projet est actuellement sur le dépôt privé GitHub suivant https://github.com/axeljunior1/bnp_projet_formatage_v3. J'ai déployé l'application sur le sur l'hébergeur Netlify à l'adresse suivante : <https://formatage.netlify.app/>.

L'image ci-dessous nous donne une présentation de la page.

Formater les articles

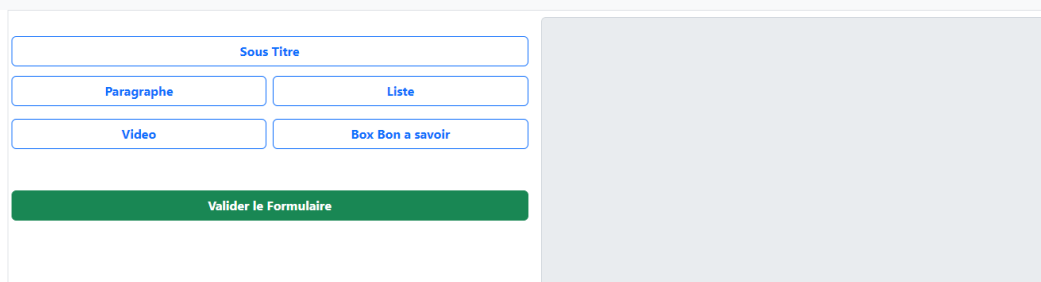


Figure 34 : résultat de la page web développé

g- Pistes d'améliorations

Le travail réalisé est parfaitement fonctionnel. Pour rester dans un esprit d'amélioration perpétuelle, je peux proposer de passer par un WYSIWYG pour parfaire le formatage, prendre en compte tous types d'éléments, et ainsi minimiser au maximum le temps de formatage.

3. Projet d'intégration d'une page web

Ce projet est une demande de mon manager. Cette demande a été formulée à partir d'une expression de besoin. Cette expression de besoin a été formulé par

Ce projet a pour objectif d'intégrer une page dont une maquette a été réalisée au préalable. Une chaîne spécifique doit être suivie pour que cette page arrive sur le site de Cardif. Les différentes équipes qui interviennent sont les suivantes :

- La Team Édito : c'est elle qui produit les contenus
- La Team UX/UI : c'est elle qui se charge de maquetter les contenus de la team Édito
- La Team Digital Factory : celle à laquelle je suis affecté. Elle gère l'intégration des maquettes de la Team UX/UI.

Je suis le développeur désigné pour réaliser l'intégration de cette page web. L'image ci-dessous montre un extrait de la maquette qui a été réalisée. Elle est également disponible à l'adresse suivante : <https://bnppcardif.invisionapp.com/public/prototype/sl7hhab0g00kqv401cwud6ikg/1664a69d>.


Cette page à elle seule va regrouper tous mes éléments du développement web Html. On va retrouver notamment :

- Les images,
- Les Icônes,
- Les Boutons,
- Les accordéons,
- Les listes,
- Les pop-up

Je travaille sur ce projet depuis le lundi 5 septembre (03 jours). J'ai réalisé à la date d'aujourd'hui deux sections complète de cette page. La difficulté a résidé dans la qualité des images, leur type (PNG, JPEG...) et leur taille. En outre, l'aspect accessibilité est d'une très grande importance car j'entame ce projet après avoir acquis de nombreuses notions en accessibilité web.

Ce projet va donc être très riche dans la mesure où il lie un grand nombre de notions web, de style et de scripts à la contrainte que pose l'accessibilité web qui prône un développement web dans sa forme la plus basique.



**CARDIF**
GRUPE BNP PARIBAS

L'assureur d'un monde qui change

Que recherchez-vous ?


ASSURANCE DE PRÊT IMMOBILIER

ASSURANCE HABITATION

PRÉVOYANCE

ÉPARGNE

RETRAITE

 Nos conseils
Prévention & Bien-être

Cardif Libertés Emprunteur,

Être bien protégé à un tarif compétitif.

Mon tarif en 1 min

 OSCAR 2022 du meilleur contrat d'assurance emprunteur



Économisez plus de 10 000€ ! ⁽¹⁾

**Nos clients parlent de nous**

 **4.5/5** | 325 avis

Lire les avis de nos clients 

Ils ont choisi d'économiser avec Cardif Libertés Emprunteur

Estelle & Thomas

**Estelle**
27 ans
Ne fume pas
Chargée d'affaires

**Thomas**
28 ans
Ne fume pas
Cadre commercial

Nous avons économisé **37 927€** ! ⁽²⁾



Leur projet : Acheter leur première maison

Crédit de 754 458€ au taux de 1,35% sur 25 ans

Voir les détails +

Proposition de leur banque

Coût de l'assurance **49 041€**
163,47€ / Mois

VS

Proposition de Cardif

Coût de l'assurance* **11 114€**
37,04€ / Mois

*3 000€ les 8 premières années

37 927€ d'économie réalisée sur 25 ans

Prêt pour un accompagnement sur mesure ?

Et si vous souhaitez changer d'assurance, nos experts s'occupent de toutes les démarches pour vous.

Être rappelé

Contacteur un expert

Figure 35 : maquette de la page des produits clés (Extrait 1)

UNIVERSITÉ
DE LORRAINE

INP
LORRAINE

Axel Junior
MENGUE FEUKENG

{ 37 }



**Un contrat accepté
par toutes les banques**

Pour assurer votre prêt avec les
mêmes **garanties** et voir plus.



**Un tarif
compétitif ***

Pour envisager d'autres projets
(travaux, voyages, voiture...).



**Des experts joignables
par téléphone**

Pour tout vous expliquer et bien
choisir vos garanties, options.



**Un accord
obtenu en 24h**

Pour disposer de votre
attestation d'assurance de prêt.

*Avec l'éventuelle économie réalisée



0€ de frais de gestion



**1 000€ d'aide de retour
à l'emploi**



2 000€ de prestation



Résiliation prise en charge



**Centres médicaux agréés
Cardif accessibles sans frais**



**Démarches simplifiées
en cas d'arrêts maladie**

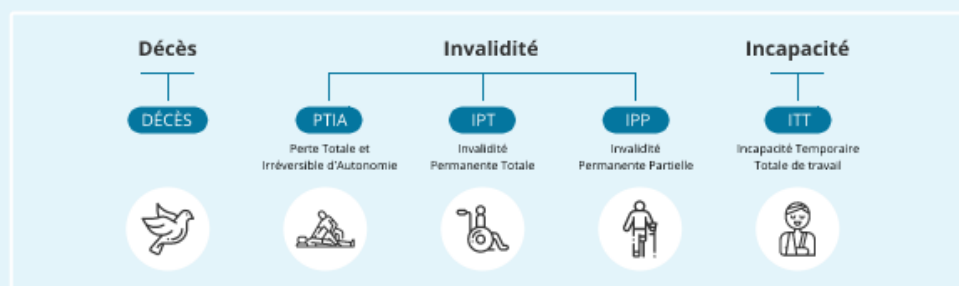


Irrévocabilité du tarif



[Comment changer d'assurance emprunteur ?](#)

Focus sur les garanties et les options de Cardif Libertés Emprunteur



Les 3 options disponibles pour renforcer votre protection :

Sérénité +

Vous êtes couvert sans condition d'hospitalisation pour les affections dorsales et psychiatriques. La souscription à cette option peut être exigée par votre banque.

Perte d'emploi

En cas de chômage, l'option perte d'emploi prend en charge, selon la quotité assurée, la moitié de vos mensualités sur une période maximum de 18 mois, consécutifs ou non après 90 jours de franchise.

Prévoyance

Elle vous permet d'avoir une couverture complémentaire en cas de décès de l'un des 2 co-emprunteurs. Cette option est possible si vous n'êtes pas assuré à parts égales sur votre contrat d'assurance de prêt.

Figure 36 : maquette de la page des produits clés (Extrait 2)

III. Difficultés rencontrées et acquis du stage

A. Difficultés rencontrées

- **Accès matériels** : la toute première difficulté que j'ai rencontrée à mon arrivée est celle du badge. Pour accéder au bâtiment de Cardif, nous avons besoin d'un badge. Malheureusement, à mon arrivée mon badge n'avait pas été commandé et n'était donc pas en cours de fabrication. Ceci est valable pour le badge de cantine et pour mon PC de travail. J'ai dû attendre courant de la semaine pour les avoir.
- **Accès logiciels** : après le matériel, nous avons dû attendre les droits logiciels. Pour les nouveaux arrivants, une automatisation des droits doit être mise en place afin de faciliter l'arrivée de ceux-ci. J'ai mis plus de 2 semaines pour avoir mes accès de télétravail et plus d'un mois pour les accès aux environnements de travail.
- Limitation du logiciel et des PC de Cardif, impossible d'installer certains environnements de travail : j'ai comme les autres membres de l'équipe rencontré des difficultés avec l'impossibilité d'installer n'importe quel autre logiciel, nous avons également l'accès sur certains sites web qui est bloqué par les administrateurs.
- Mauvaise gestion des environnements de travail, bug et pannes incessante sur les applications : Cardif utilise aujourd'hui le CMS Liferay pour la gestion de ses contenus. Toutefois, l'interface utilisateur est très souvent inaccessible, les requêtes en GET ou en POST qui nous renvoie des pages blanches. On va aussi noter le rechargement instantané des pages en cours d'utilisation. Ces pannes rendent très difficile le travail du développeur que je suis.

Concernant les projets sur lesquels j'ai travaillé, on va noter 2 difficultés particulières :

- Problème du poids des images dans les pages web et le format des images qui entre en opposition avec les valeurs du développeur dans la mesure où il doit penser à la maintenance du système qu'il conçoit : le poids des images compte pour beaucoup dans la vitesse du changement d'une page web. Cardif a décidé de réduire au maximum la taille des images du site. Pour mon projet d'intégration de la page web, je me devais de faire très attention au poids des images et à leur format car le format va influencer la taille de l'image et certains formats d'images sont plus adaptés que d'autres en fonction des situations. Je dois prendre ceci en compte tout en gardant à l'esprit l'aspect évolution et modification de la page web que je code.
- Enfin le plus gros souci auquel j'ai dû faire face : les composants contrôlés et les composants non contrôlés : comme je l'ai expliqué plus haut, afin de mener à bien le projet d'optimisation du temps de formatage, j'ai dû creuser au plus profond de la structure de React afin de me rendre à l'évidence que seuls les composants non contrôlés peuvent résoudre mon problème. Ça revenait à essayer de contrôler un composant dont on ne connaît ni sa date de naissance future ni s'il va naître dans le futur.



B. Les acquis au terme du stage

Au terme de ce stage, j'ai pu :

- Découvrir ce qu'est le métier de Content Manager dans la vie d'un site web,
- Travailler avec tous les profils métiers du digital autour de la vie d'un site : webmasters, chef de projet digital, UX designers, rédacteurs de contenus,
- Comprendre comment fonctionne le métier de téléconseiller et celui d'experts en acquisition de trafic au sein de Cardif,
- Apprendre à intégrer des contenus digitaux dans différents outils de gestion de contenus,
- Acquérir de nombreuses notions comme le SEO (référencement naturel Google),
- Acquérir de nombreuses notions de base du Framework React JS, de la bibliothèque jQuery,
- Améliorer grandement mes connaissances de Html, Css et Bootstrap autant sur des notions globales telles que les accordéons, les carrousels, les cards, les formulaires, les modales et bien d'autres, que sur des notions particulières telles que la gestion des boutons, des images, des pseudo-éléments ou des événements en JavaScript,
- Monter en compétente de façon fulgurante sur le concept qu'est l'accessibilité web,
- Acquérir des bonnes pratiques lors du développement en Html.

Conclusion

Ainsi, dans le cadre de mes études d'ingénieurs dans les systèmes d'information et réseaux, j'ai eu le plaisir d'effectuer un stage au sein du leader mondial en assurance des emprunteurs qu'est BNP Parisbas Cardif. Cette expérience s'est avérée extrêmement enrichissante car elle m'a permis de découvrir en plusieurs métiers qui interviennent dans la vie d'un site web, ainsi que les différentes missions et postes qui contribuent à l'édition du contenu et à sa mises à jour. On a d'ailleurs pu s'apercevoir que l'édition d'un site web peut dépendre aussi bien des facteurs internes (arrivée d'un nouveau produit en assurance, ou sur simple décision du comité éditorial) qu'externe (je pense notamment à la loi de 2005 qui oblige tous les services numériques publics à être accessible à tout citoyen), d'où le projet d'accessibilité sur lequel j'ai travaillé.

En participant aux différentes évolutions et aux différents projets au sein de l'entreprise, j'ai pu mettre en œuvre une partie de mes connaissances théoriques, de mon savoir-faire et de mon savoir être. J'ai en retour pu acquérir un très large éventail de connaissances et spécifiquement dans les technologies de base du développement web dans un premier temps et puis j'ai découvert ce qu'est l'accessibilité web dans un projet où j'ai travaillé avec un leader de l'accessibilité numérique en France. En outre, j'ai identifié un problème, je l'ai remonté à mon manager, j'ai convaincu le manager d'en faire un projet et de me laisser la charge de celui-ci. J'ai travaillé dessus en parallèle et je l'ai mené à bien. j'ai résolu un problème qui existe qui aurait continué si ma solution n'avait pas été proposé.

Aussi bien que travailler chez Cardif sur différents projets et évolutions tels que l'intégration des pages web et l'accessibilité numérique m'a permis de perfectionner ma maîtrise du Html, Css et JavaScript, le projet d'optimisation du temps de formatage des articles m'a inculqué de bases solides en React JS. Ceci a accentué mon amour pour les technologies de développement des interfaces utilisateur et j'aimerais dorénavant m'orienter vers un développement web à majorité front-end.

Annexes

Annexe 1 : Liens

Lien 1 : <https://document-information-cle.cardif.fr/accessibilite>

Lien 2 : <https://developer.mozilla.org/fr/docs/Web/HTML/Element/hr>

Lien 3 : <https://fr.reactjs.org/docs/forms.html#alternatives-to-controlled-components>

Lien 4 : <https://fr.reactjs.org/docs/uncontrolled-components.html>

Lien 5 : <https://www.numerique.gouv.fr/publications/rgaa-accessibilite/#contenu>

Annexe 3 : Table des illustrations

Figure 1 : Organigramme direction E-Business	5
Figure 2 : organigrammes des différentes équipes de la E-Business	6
Figure 3 : localisation du siège de BNP Paribas Cardif	7
Figure 4 : actualité Altixia	11
Figure 5 : étapes de préparation du visuel de publication	12
Figure 6 : extraits d'offre immobilière de la gamme financière	13
Figure 7 : emplacement des documents de gestion délégués	14
Figure 8 : les fonds de la gamme Cardif.	14
Figure 9 : les nouveaux fonds de la gamme Cardif.....	15
Figure 10: les fonds des mois précédents.	15
Figure 11: exemple d'article publié sur le site cardif.fr	16
Figure 12 : exemple d'actualité publiée sur le site des partenaires.....	17
Figure 13 : mise à jour d'une bannière sur le site des partenaires	17
Figure 14 : modification d'une LP dans le but de la rendre plus accessible et pour optimiser de l'espace	18
Figure 15 : mise à jour d'une LP suite à l'ajout de nouveaux produits.....	18
Figure 16 : accessibilité sur la balise de décoration.....	21
Figure 17: accessibilité sur la balise <hr> de décoration.....	22
Figure 18 : classes nécessaires pour corriger l'accessibilité des balises 'hr'	22
Figure 19 : ensemble de classes pour rendre accessible le soulignement.....	22
Figure 20 : accessibilité label/champ d'un formulaire	23
Figure 21 : accessibilité sur un formulaire avec la balise fieldset	23
Figure 22 : création du composant liste	26
Figure 23 : code du box d'éléments	27
Figure 24: résultat du box de bouton.....	28
Figure 25: rendu de app.js.....	29
Figure 26: constructeur de la classe App.....	30
Figure 27: code exécuté dans la fonction choix_btn pour le clic sur sous-titre	30
Figure 28 : fonction choix_btn pour un choix de valider le formulaire.....	31
Figure 29 : fonction de formatage des paragraphes	32
Figure 30 : fonction de formatage des sous-titres	33
Figure 31 : fonction de formatage des sous-titres	33
Figure 32 : fonction de formatage du box « bon à savoir »	34
Figure 33 : fonction de formatage iframe vidéo	34
Figure 34 : résultat de la page web développé	35
Figure 35 : maquette de la page des produits clés (Extrait 1).....	37
Figure 36 : maquette de la page des produits clés (Extrait 2).....	38
Figure 37: accessibilité sur les accordéons.....	44
Figure 38 : code obtenu pour le formatage d'un sous-titre et ses paragraphes	44
Figure 39 : résultat du formatage d'un sous-titre + paragraphe.....	45

Annexe 3 : Images



Les garanties de votre contrat d'assurance habitation

La garantie dégât des eaux	✓
La garantie incendie	^
La garantie incendie permet la protection du logement, des biens personnels et des dommages corporels causés par le feu. La garantie explosion (dégâts occasionnés par une explosion ou une implosion) est souvent incluse dans la garantie incendie.	
La garantie bris de glace	✓
La garantie vol	✓
La garantie catastrophes naturelles	✓
La garantie événements climatiques (grêle, neige, tempête)	^
La garantie attentats	✓
La responsabilité civile privée	✓
L'assurance scolaire et extra-scolaire	✓

Figure 37: accessibilité sur les accordéons

Formater les articles

Sous titre

Formatage des sous titres

1

Supprimer

Paragraphe

une question d'esthétique et de prévoyance, on retire les espaces en double puis les espaces en début et en fin de phrase.
On peut voir en annexe à quoi ressemble une ensemble sous-titre + paragraphe sur le site de Cardif (Voir annexe 3) et comment se présente le code formaté avec la page web conçu (Voir annexe 3).

Supprimer

Sous Titre

Paragraphe

Liste

Video

Box Bon a savoir

Valider le Formulaire

```

<div class="d-flex">
  <span class="title-number h3 text-fuchsia z-depth-5">
    1
  </span>
  <h2 class="text-primary"id="formatage-des-sous-titres">
    Formatage des sous titres
  </h2>
</div>

<p>
  l'idée est de récupérer un sous-titre et son numéro afin de renvoyer le code html
  qui correspond à la syntaxe de code des sous-titre au sein de la Digital Factory (DF).
  Au sein de la DF, dans un article, tout sous-titre doit avoir un identifiant. Pendant le
  formatage des sous-titre, il faut ajouter cet identifiant unique en attribut.
</p>

<p>
  L'identifiant correspond au titre auquel on retire les accents, on remplace les
  espaces par les tirets et on retire tout caractère qui n'est pas alphanumérique. Pour
  une question d'esthétique et de prévoyance, on retire les espaces en double puis les
  espaces en début et en fin de phrase.
</p>

<p>
  On peut voir en annexe à quoi ressemble une ensemble sous-titre + paragraphe
  sur le site de Cardif (Voir annexe 3) et comment se présente le code formaté avec la
  page web conçu (Voir annexe 3).
</p>

```

Figure 38 : code obtenu pour le formatage d'un sous-titre et ses paragraphes



1 Formatage des sous titres

L'idée est de récupérer un sous-titre et son numéro afin de renvoyer le code html qui correspond à la syntaxe de code des sous-titre au sein de la Digital Factory (DF). Au sein de la DF, dans un article, tout sous-titre doit avoir un identifiant. Pendant le formatage des sous-titre, il faut ajouter cet identifiant unique en attribut.

L'identifiant correspond au titre auquel on retire les accents, on remplace les espaces par les tirets et on retire tout caractère qui n'est pas alphanumérique. Pour une question d'esthétique et de prévoyance, on retire les espaces en double puis les espaces en début et en fin de phrase.

On peut voir en annexe à quoi ressemble une ensemble sous-titre + paragraphe sur le site de Cardif (Voir annexe 3) et comment se présente le code formaté avec la page web conçu (Voir annexe 3).

Je veux constituer mon épargne

Cardif vous propose d'être rappelé par un expert indépendant*.

C'est simple et sans engagement.

Être contacté par un expert

*Les experts sont des conseillers en gestion de patrimoine indépendants, partenaires de CARDIF.

Figure 39 : résultat du formatage d'un sous-titre + paragraphe



Annexe 3 : Table des abréviations

Abréviations	Sigle
DF	Digital Factory
JS	JavaScript
E-B	E-Business
DOM	Document Object Model
CMS	Content Management System
OPC	Organisme de Placement Collectif
UI	User Interface