

MEMOIRE



IPSSI

ÉCOLE SUPÉRIEURE PRIVÉE D'INFORMATIQUE ET DU NUMÉRIQUE

JAFFREZOU LUDOVIC

2021

Table des matières

Remerciements	3
Glossaire	4
Présentation du Contexte	10
Mes différentes missions	11
1 ^{ère} Mission : Le Décret Tertiaire	14
2 ^{ème} Mission : Dyson API	17
3 ^{ème} Mission : API Température	19
4 ^{ème} Mission : QR Code	22
5 ^{ème} Mission : Map Maquette	24
Analyse de la situation	26
Mise en place de solutions et actions	28
Analyse des résultats	30
Conclusion	31
Références	32

Problématique : Comment gérer l'évolution d'une plateforme de gestion de données smart building ?

Remerciements

Je souhaite remercier l'entreprise « Nextiim » qui m'a accueilli afin d'effectuer mon alternance et plus particulièrement Yann Jobert et Julien Cousin qui ont su m'accompagner et me permettre de monter en compétences dans le développement web par les différentes missions qui m'ont été confiées et qui pour la plupart m'ont permis de découvrir de nouveaux langages et de nouvelles technologies.

J'ai été très bien accueilli et je me suis senti à l'aise tout au long de mon alternance. Travailler dans les trois entreprises m'a permis de m'épanouir et d'être d'autant plus impliqué dans le projet « Sextant » afin de contribuer à son maintien et son évolution.

Ainsi j'ai pu prendre confiance en moi et gagner en autonomie. J'ai pu prendre des décisions quant à la gestion des projets et les propositions d'améliorations et de créations de nouveautés qui ont toujours été écoutées et analysées afin d'en juger la pertinence.

Le fait de pouvoir exprimer son avis et être force de proposition est selon moi la meilleure philosophie que l'on peut être amenée à rencontrer dans une entreprise. J'apprécie que Nextiim soit une « petite » structure et me permette de suivre le projet de A à Z et de m'investir au maximum. En effet, le projet m'a très vite intéressé et a été très formateur dans mon domaine d'activité.

Glossaire

CSS : Les feuilles de style en cascade, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML.

Immotique : La domotique est l'ensemble des techniques de l'électronique, de physique du bâtiment, d'automatisme, de l'informatique et des télécommunications utilisées dans les bâtiments, plus ou moins « interopérables » et permettant de centraliser le contrôle des différents systèmes et sous-systèmes de la maison et de l'entreprise.

API : Une API est un ensemble de définitions et de protocoles qui facilite la création et l'intégration de logiciels d'applications. API est un acronyme anglais qui signifie « Application Programming Interface », que l'on traduit par interface de programmation d'application.

Smart building : Un « smart building », ou bâtiment intelligent, c'est bien plus qu'un bâtiment traditionnel. Pourquoi ? Parce qu'il intègre une couche d'outils numériques permettant de nombreux avantages, qui vont de l'optimisation des ressources énergétiques au confort des collaborateurs. Pour les entreprises, c'est la possibilité de diminuer les coûts opérationnels, tout en proposant à leurs équipes les meilleures conditions de travail possibles.

HTTP : L'HyperText Transfer Protocol est un protocole de communication client-serveur développé pour le World Wide Web. HTTPS est la variante sécurisée par l'usage des protocoles Transport Layer Security. HTTP est un protocole de la couche application.

GTB : La gestion technique de bâtiment est un système informatique généralement installé dans des grands bâtiments ou dans des installations industrielles afin de superviser l'ensemble des équipements qui y sont installés.

JQuery : jQuery est une bibliothèque JavaScript libre et multiplateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML des pages web. La première version est lancée en janvier 2006 par John Resig.

Forge Autodesk : Application de rendu des données de modèle 3D et 2D dans un navigateur. Les modèles peuvent provenir d'un large éventail d'applications telles qu'AutoCAD, Fusion 360, Revit et bien d'autres.

OVH : OVHcloud, anciennement OVH, est une entreprise française. Elle pratique initialement de l'hébergement de serveur, et est un fournisseur d'accès à Internet (FAI), puis opérateur de télécommunications pour les entreprises. Elle se développe, à la fin des années 2010, dans le cloud computing (informatique en nuage).

ThingWorx : La plateforme **ThingWorx** de PTC est la première plateforme de développement d'applications conçue pour les besoins de l'**Internet des objets** et du monde connecté, un monde caractérisé par une explosion de capteurs connectés, d'appareils et d'équipements. ThingWorx permet aux industries de créer à moindre coût de la valeur à partir de leurs produits ou moyens de production connectés, et de fournir des solutions complètes. Cette solution offre un environnement complet de conception, d'exécution et d'intelligence applicatives : plateforme moderne et complète, déploiement agile dix fois plus rapide, approche basée sur la **collecte de vos données**, leur **valorisation** et leur **restitution** par nos experts, quel que soit la nature de vos outils et machines, vous permettant de dégager un **ROI** rapidement.

Matterport : Matterport crée un jumeau digital réaliste pour tous les types d'espaces bâtis, permettant aux clients et aux parties prenantes d'explorer et d'évaluer facilement des biens.

IOT : L'Internet des objets ou IdO est l'interconnexion entre l'Internet et des objets, des lieux et des environnements physiques. L'appellation désigne un nombre croissant d'objets connectés à Internet permettant ainsi une communication entre nos biens dits physiques et leurs existences numériques.

Wireshark : Wireshark est un analyseur de paquets libre et gratuit. Il est utilisé dans le dépannage et l'analyse de réseaux informatiques, le développement de protocoles, l'éducation et la rétro-ingénierie.

TCP/IP : TCP/IP est une suite de protocoles. Le sigle TCP/IP signifie « Transmission Control Protocol/Internet Protocol » et se prononce «T-C-P-I-P». Il provient des noms des deux protocoles majeurs de la suite de protocoles, c'est-à-dire les protocoles TCP et IP).

TCP/IP représente d'une certaine façon l'ensemble des règles de communication sur internet et se base sur la notion adressage IP, c'est-à-dire le fait de fournir une adresse IP à chaque machine du réseau afin de pouvoir acheminer des paquets de données.

MQTT : MQTT (Message Queuing Telemetry Transport²) est un protocole de messagerie publish-subscribe basé sur le protocole TCP/IP.

Il a été initialement développé par Andy Stanford-Clark (IBM) et Arlen Nipper (EuroTech). Il est conçu pour les connexions avec des sites distants où la bande passante du réseau est limitée.

Urn : Uniform Resource Name (URN), traduit littéralement de l'anglais par « nom uniforme de ressource », est le nom d'un standard informatique dans le domaine de l'Internet qui concerne principalement le World Wide Web.

Le document de base est la RFC 2141 URN Syntax publiée en 1997 par l'Internet Engineering Task Force. Elle donne une syntaxe de chaîne de caractères utilisable pour identifier une ressource (un document, une image, un enregistrement sonore, etc.) globalement, durant toute son existence, indépendamment de sa localisation ou de son accessibilité par Internet. Les Uniform Resource Names sont des Uniform Resource Identifiers (URI) dont la méthode est urn.

GraphQL : GraphQL¹ (pour Graph Query Language) est un langage de requêtes et un environnement d'exécution, créé par Facebook en 2012, avant d'être publié comme projet open-source en 2015². Inscrit dans le modèle Client-Serveur, il propose une alternative aux API REST¹. La requête du client définit une structure de données, dont le stockage est éventuellement distribué, et le serveur suit cette structure pour retourner la réponse³. Fortement typé, ce langage évite les problèmes de retour de données insuffisants (under-fetching) ou surnuméraires (over-fetching).

BingMaps : Bing Cartes (Bing Maps, auparavant Live Search Maps, Windows Live Maps et Windows Live Local) est un service web de cartographie faisant partie du moteur de recherche Bing et utilisant le service de Bing Maps for Enterprise.

HTTPS : L'HyperText Transfer Protocol Secure (HTTPS, littéralement « protocole de transfert hypertextuel sécurisé ») est la combinaison du HTTP avec une couche de chiffrement comme SSL ou TLS.

HTTPS permet au visiteur de vérifier l'identité du site web auquel il accède, grâce à un certificat d'authentification émis par une autorité tierce, réputée fiable (et faisant généralement partie de la liste blanche des navigateurs internet). Il garantit théoriquement la confidentialité et l'intégrité des données envoyées par l'utilisateur (notamment des informations entrées dans les formulaires) et reçues du serveur. Il peut permettre de valider l'identité du visiteur, si celui-ci utilise également un certificat d'authentification client.

Introduction

Le sujet de mon mémoire est la gestion de bâtiments “smart building” à l’aide d’un outil de supervision permettant d’optimiser les dépenses énergétiques d’un bâtiment ainsi que de contrôler l’état des équipements connectés.

Les bâtiments dit « smart-buildings » sont des bâtiments qui sont interconnectés par des capteurs à des serveurs de collecte de données, traitant ainsi les données qui par la suite ont pour objectif de mieux contrôler et gérer la consommation énergétique d’un bâtiment.

Ce concept aussi appelé Immotique a vu le jour dans les années 1990. Il désigne l’ensemble des systèmes domotiques établis dans un bâtiment à grande échelle comme un immeuble, un site industriel ou encore un hôpital.

Par l’immotique notre outil permet la Gestion Technique de Bâtiments (GTB), il permet au gestionnaire de site d’avoir une vue globale sur le fonctionnement des équipements de son bâtiment ou de ses bâtiments.

Les données présentées peuvent être de plusieurs origines :

- Des mesures (températures/consommations/temps de fonctionnement)
- Des états (allumé/éteint)
- Des alarmes (pannes /anomalies/absurdité des données)

Notre outil de supervision est donc un ensemble de fonctionnalités regroupées dans une interface mise à disposition du client.

Aujourd’hui notre solution est plus qu’utile pour les entreprises car elle permet une gestion de ses bâtiments “smart building” concentrée sur un site.

J’ai été amené à développer plusieurs fonctionnalités durant mon alternance en lien avec notre plateforme « Sextant » afin de la mettre en adéquation avec les différents besoins des clients.

Les différents projets et fonctionnalités seront présentés tout au long de ce mémoire par des exemples et explications.

Présentation de l'entreprise



J'ai réalisé mon alternance dans l'entreprise Nextiim, il s'agit d'une startup qui travaille en lien avec deux autres entreprises « Emovision » et « Envie-Energie ».

Nextiim étant l'entreprise de « développeurs », « Emovision » la partie techniciens/administrateurs réseaux et « Envie-Energie » l'entreprise en charge de la gestion de la partie énergétique et commerciale.

Ces trois entreprises sont complémentaires et cumulent environ 30 personnes.

Nextiim est une entreprise de développement de logiciels de gestion de bâtiments “smart-building”.

J'ai été amené à travailler sur notre plateforme appelé « Sextant » afin de modifier/implémenter et d'améliorer des fonctionnalités qui seront présentées tout au long de ce mémoire.

Cette plateforme repose sur le logiciel ThingWorx de l'entreprise PTC. Il s'agit d'un logiciel IOT permettant le développement d'applications conçues pour les besoins de l'Internet des Objets et du monde connecté.



Concrètement le choix de l'utilisation de ThingWorx par Nextiim représente un énorme gain de temps du fait des fonctionnalités qu'elle propose. Elle est conçue pour que les développeurs puissent concevoir au mieux leur environnement.

Présentation du Contexte

Aujourd'hui le domaine d'activité de « Nextiim » est un domaine porteur et d'avenir car la loi Elan impose aux entreprises une obligation de déclaration annuelle des consommations des bâtiments tertiaires ainsi qu'un affichage de leur consommation.

Cette loi vise à réduire de 40% les consommations d'ici 2030, 50% en 2040 et 60% en 2050 par rapport à une consommation de référence, en fonction des critères du bâtiment.

Notre plateforme « Sextant » prend donc à ce jour toute son importance pour les entreprises possédant des bâtiments tertiaires afin de leur permettre de contrôler leurs consommations.

Le « smart-building » est donc le cœur de notre domaine d'activité et les outils que propose notre entreprise sont toujours en lien avec ce domaine.

J'ai intégré « Nextiim » en Octobre 2020 en tant que développeur web avec l'envie d'apporter ma pierre à l'édifice dans ce projet que m'a exposé Julien Cousin durant ma phase de recrutement.

J'ai aussi assez vite constaté que le projet était très bien conçu, très attrayant et très novateur alors qu'il n'a été réalisé à la base qu'avec deux développeurs, ce qui m'a encore plus donné envie de me joindre au projet.

Il m'a fallu quelques temps avant de bien comprendre toute l'architecture et la conception des différents projets présents au sein de l'entreprise

Mais ma montée en compétence a été d'autant plus importante qu'un autre alternant était présent dans le développement, ce qui nous a permis de nous entraider durant les différents projets qui nous avons été amenés à réaliser.

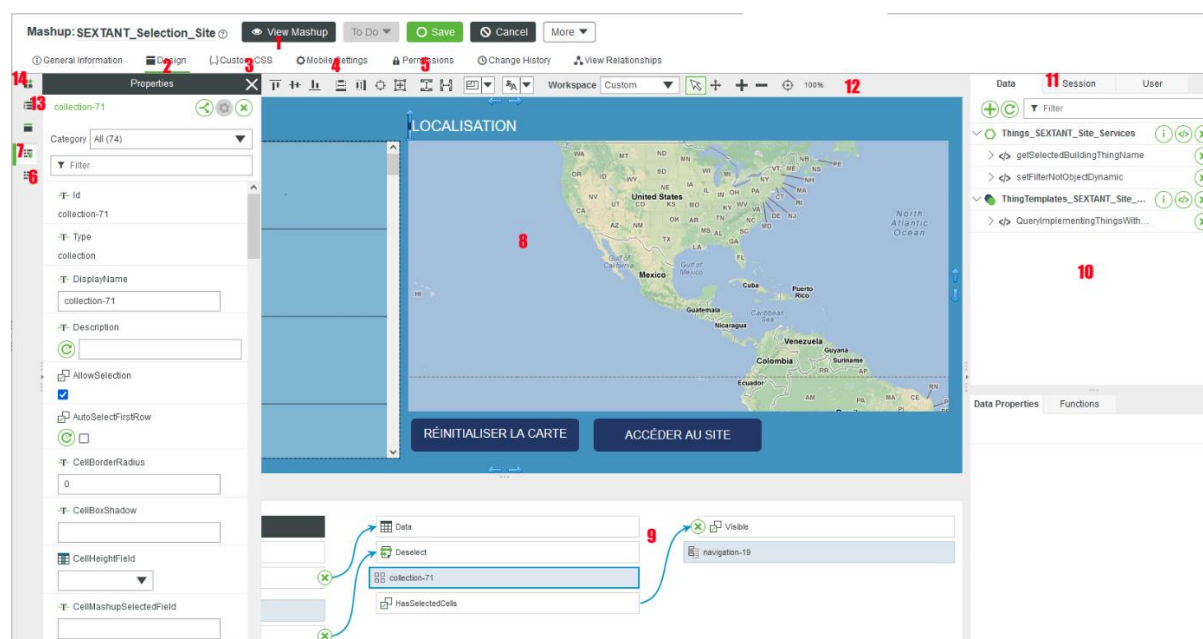
Durant mon alternance, différentes missions m'ont été confiées. Certains des projets étaient préexistants à mon arrivée dans l'entreprise. Le but a donc été de les améliorer/optimiser et d'implémenter de nouvelles fonctionnalités. En revanche, pour d'autres projets il n'y avait aucune base et j'ai dû les développer en intégralité, ce qui a été très intéressant car j'ai pu tout concevoir, créer et mettre en place moi-même.

Mes différentes missions

Il m'a été confié plusieurs missions visant à améliorer notre plateforme « Sextant ».

Tout d'abord j'ai été amené à prendre en main la plateforme, ce qui a demandé un certain temps, ainsi que son architecture afin que je puisse développer et améliorer des fonctionnalités.

Afin d'expliquer plus en détail notre outil, nous allons voir son fonctionnement mais plus particulièrement la création de mashup.



Le screen ci-dessus représente un « Mashup » de configuration de l'interface web qui sert à la connexion et sélection d'un bâtiment d'un client.

Nous allons en faire une présentation plus détaillée afin de comprendre le fonctionnement de notre outil.

1 : Cet onglet permet de visualiser la version web et le résultat du courant « Mashup » aussi appelé vue.

2 : C'est sur cette partie que nous allons gérer la partie Design de notre vue, la disposition des différents éléments / widgets que nous allons utiliser afin de créer notre interface.

3 : Nous avons la possibilité de modifier le CSS (Feuilles de Style) dans notre vue ce qui permet d'ajuster la mise en forme des différents éléments présents sur notre page Web.

4 : Nous avons aussi la possibilité de configurer notre mashup pour le rendre compatible à l'utilisation sur mobile, de part des paramètres comme la gestion dynamique de la hauteur et largeur de l'affichage.

5 : L'onglet « permissions » est un onglet très utile car c'est celui qui va nous permettre de gérer les permissions des différents utilisateurs afin de modifier et lire les mashups. Cela permet une meilleure gestion de l'isolation des différents clients présents sur notre plateforme afin que chaque client puisse interagir uniquement avec les bâtiments qu'il possède.

6 : C'est ici que nous gérons la première couche de design des différents widgets présents sur notre mashup, c'est un peu comme l'onglet custom CSS mais celui-ci est applicatif à chaque widget.

7 : L'outil des propriétés est sûrement celui qui est le plus utile dans la configuration de mashups car il va nous permettre de configurer de manière ciblée les différentes propriétés que les widgets proposent. Par exemple comme nous pouvons le voir dans le screen ci-dessus, nous avons la possibilité d'autoriser l'utilisateur à sélectionner un bâtiment à travers notre widget de collection qui répertorie tous les bâtiments d'un client.

8 : Nous pouvons voir ici l'onglet de prévisualisation, c'est à partir de là que nous pouvons visualiser les modifications que nous faisons en direct sur le mashup.

9 : Aussi appelé l'onglet de « Bindings » c'est ici que nous allons gérer les différentes interactions entre les widgets, le design et tout autre évènement intervenant sur notre mashup. Par exemple, lorsque l'utilisateur clique sur son bâtiment, nous le redirigeons sur les différents éléments de son bâtiment comme le récapitulatif de ses consommations énergétiques sur le compteur général électrique.

10 : L'onglet de gestion de services est l'endroit où nous allons développer de manière plus personnalisée les interactions entre les widgets et le Design.

Il s'agit ici encore d'une surcouche comme présentée ci-dessus avec le CSS mais dans ce cas le service est un composant exécutant du code Javascript côté client permettant de gérer principalement les données par un langage de programmation.

11 : L'onglet de Session va, lui contenir les données d'un client connecté et est propre à chaque utilisateur. C'est ici que nous allons identifier un client précisément et savoir à quel bâtiment il a accès ainsi que les permissions et droits d'accès sur notre plateforme. Notre plateforme propose différents rôles à ses utilisateurs comme « Gestionnaire de sites », « Techniciens » ou encore « Propriétaire ». Nous reviendrons sur cette partie plus en détail ultérieurement.

12 : Enfin, la barre d'outils est certainement un des onglets les moins utilisés mais va permettre de personnaliser les différents éléments tout comme le CSS.

Ce qu'il faut comprendre avec cette plateforme c'est qu'il s'agit de notre outil principal afin de développer et d'intégrer tous nos outils, il s'agit donc du cœur de notre logiciel.

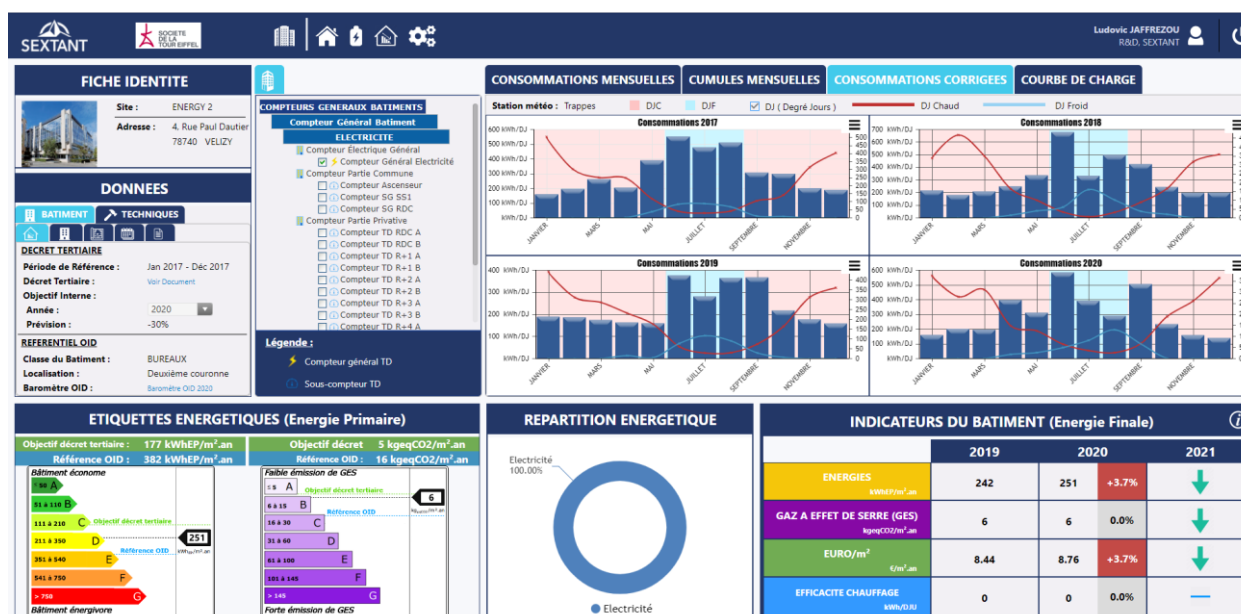
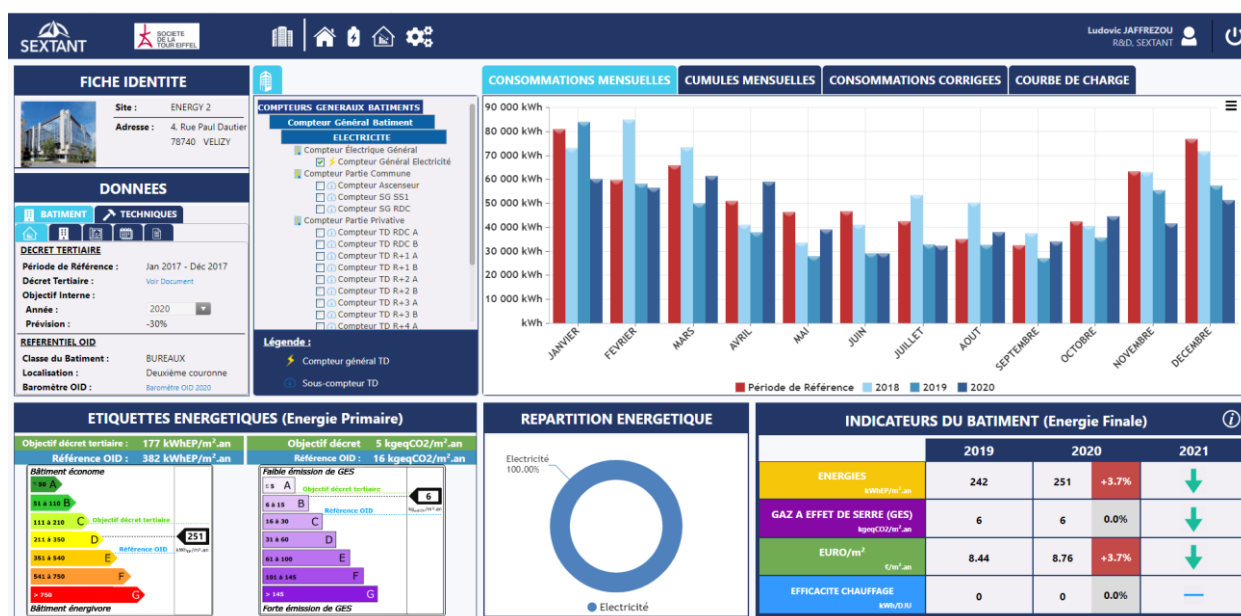
De plus, la compréhension de cette interface est primordiale car c'est la plus utilisée au sein de nos projets. Elle permet l'intégration et le paramétrage de tous nos projets annexes quant à leur utilisation dans ThingWorx.

Maintenant que l'interface de « ThingWorx » est un peu plus détaillée nous allons parler des différentes missions et fonctionnalités qui m'ont été confiées durant mon alternance.

1^{ère} Mission : Le Décret Tertiaire

Le décret tertiaire est une des plus grosses parties de « Sextant », il s'agit d'une interface d'analyse générale de la consommation d'un bâtiment sur 12 mois glissant en fonction d'une période de référence critique.

Voici un exemple du résultat que le client va avoir en arrivant sur cet onglet. Nous avons regroupé plusieurs informations quant à sa consommation.



Sur cette mission, j'ai travaillé sur la partie graphique et gestion de données pour l'affichage des « Consommations mensuelles », « Cumulés Mensuelles », « Consommations corrigées » et « Courbe de charge ».

Le décret tertiaire est un « Mashup » créé à l'aide de « ThingWorx ». D'autres projets, qui vont suivre ont été conçus « from scratch » c'est-à-dire de A à Z et n'auront pas forcément le même design.

J'ai aussi été amené à remanier l'arborescence de gestion de compteur, permettant de choisir sur quel compteur se baser pour effectuer l'ensemble des calculs ainsi que la récupération des données et affichage.

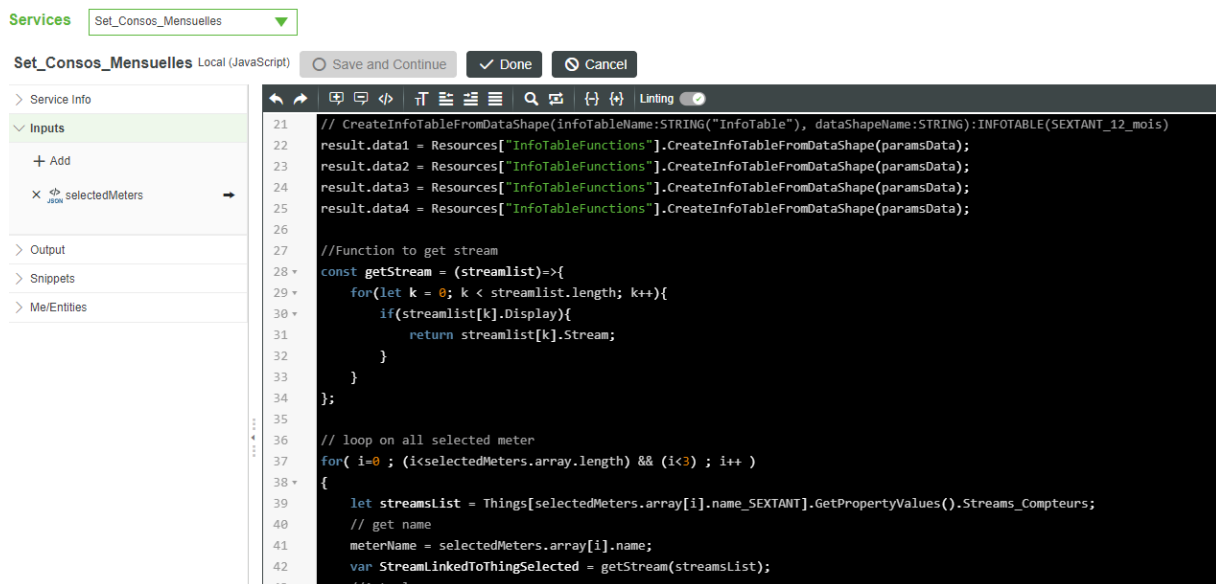
Afin de pouvoir répondre à ces missions j'ai dû créer un ensemble de « Services » développé en JavaScript permettant la récupération, le traitement de données puis un rendu conforme au résultat souhaité.

Dans cette partie j'ai pu rechercher et comprendre par moi-même les calculs énergétiques pour les différents cas. Cela m'a permis une meilleure compréhension du domaine énergétique que proposait notre plateforme.

Afin de pouvoir tracer les courbes et traiter les données nous avons dû nous baser sur des formules de calculs comme le calcul du DJU (Degré Jour Unifié) :

$$\text{DJU} = 18 - (\text{Tmin} + \text{Tmax}) / 2$$

Nous avons présenté précédemment explicitement les widgets et fonctionnalités que proposent « ThingWorx » mais voici un exemple d'un service permettant le fonctionnement du Décret Tertiaire :



Ici nous pouvons voir un service que j'ai développé permettant la récupération de données stockées par « ThingWorx », un traitement de ces données puis une sortie en affichage pour le rendu utilisateur par la suite.

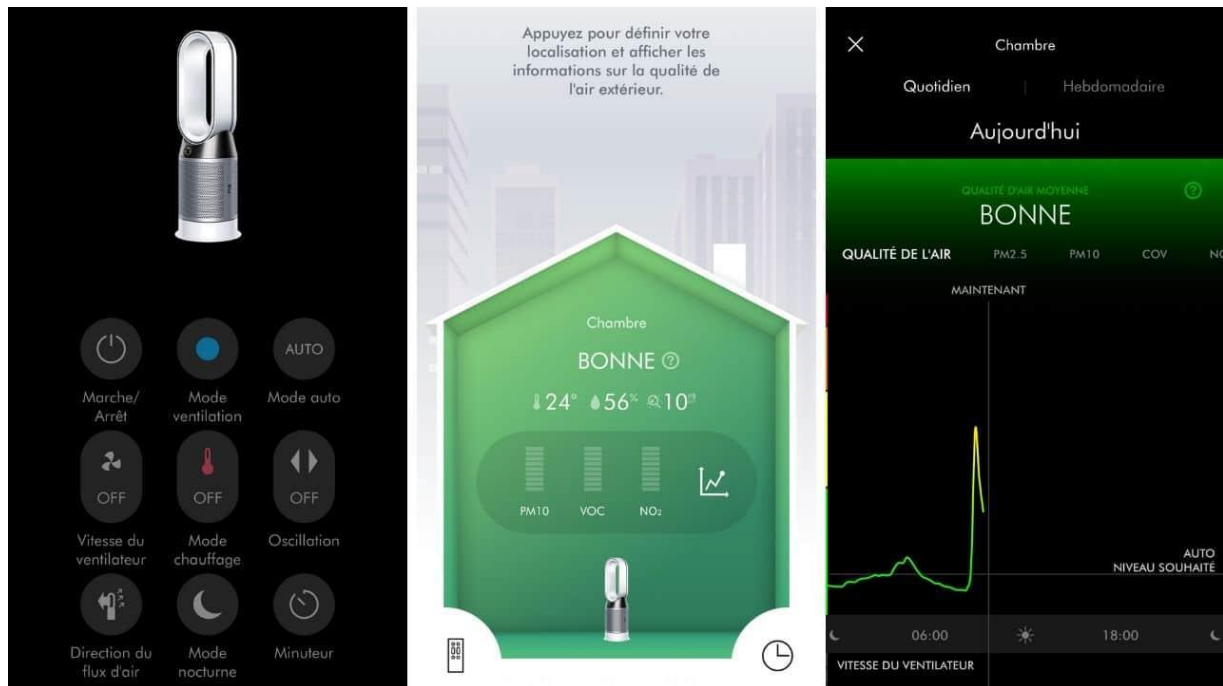
Comme évoqué précédemment dans le contexte de la **loi Elan**, l'objectif vise à réduire les consommations énergétiques des bâtiments, c'est pourquoi notre onglet de Décret tertiaire prend toute son utilité !

Le rôle du développement de ce projet permet aux entreprises de gérer leur bâtiment et de pouvoir anticiper cette loi grâce aux différentes analyses que nous leur proposons sur les données d'équipements présents dans leurs bâtiments.

2^{ème} Mission : Dyson API

Comme son nom l'indique ce projet qui m'a été confié avait pour but d'interagir avec des équipements Dyson, afin de pouvoir installer chez nos clients des appareils Dyson comme des purificateurs d'air et de les gérer à distance.

Le but étant de reproduire l'application Dyson ainsi que leur télécommande afin de pouvoir recréer les mêmes interactions pour interagir avec des équipements depuis « Sextant ».



Il s'agit d'une des premières missions qui m'a été confiée. Je me suis donc, dans un premier temps, familiarisé avec l'application créée par Dyson pour m'en inspirer et la reproduire dans notre logiciel.

J'ai donc réalisé une application permettant d'interagir avec les équipements Dyson présents au sein de nos locaux. Pour cela j'ai dû utiliser le protocole de communication MQTT, il s'agit d'un protocole de communication TCP/IP qui permet l'échange de données entre serveurs et équipements.

Dyson n'ayant pas fourni de documentation pour les développeurs, j'ai dû créer une application en récupérant les trames de communications des équipements sur notre réseau à l'aide d'un sniffer « Wireshark », outil utilisé pour analyser les « paquets de données » qui transitent sur le réseau.

A l'aide des données récupérées dans les paquets, j'ai pu créer une application envoyant les mêmes types de données sur le réseau de l'application Dyson, pour refaire les mêmes procédures que Dyson.

Ce projet avait pour but la mise en place d'un contrat avec Dyson afin de déployer des équipements Dyson comme des purificateurs d'air chez nos clients.

Etant donné que cette mission a été réalisée et développée avec les « moyens du bord », certaines des techniques utilisées n'étaient de ce ne fait pas commercialisables.

C'est pourquoi cette mission a été mise en suspens car d'autres projets plus importants ont été priorités.

3^{ème} Mission : API Température

Il m'a été confié comme mission de créer une interface et un outil permettant d'afficher en temps réel des indicateurs techniques de bâtiments sur une maquette 3D, comme la température ou l'humidité d'un bâtiment, en fonction des différents capteurs mis à disposition à travers le bâtiment.



Comme vous pouvez le voir sur le screen ci-dessus nous récupérons la maquette 3D d'un bâtiment d'un de nos clients et nous implémentons des petites popup regroupant les différentes informations récupérées sur les capteurs.

Le modèle de la maquette 3D nous est fourni par un prestataire extérieur, spécialisé dans le maquetage 3D, ensuite nous récupérons l'adresse urn générée de la maquette pour l'exploiter par la suite.

L'adresse urn étant un ID pour identifier une ressource précise, dans notre cas pour utiliser la bonne maquette.

Les technologies utilisées sont Forge Autodesk et NodeJS. Forge va nous permettre de récupérer le visuel 3D de la maquette grâce à son urn et NodeJS est une plateforme logicielle en Javascript qui permet plusieurs fonctionnalités comme :

- Sa rapidité d'exécution
- Le déploiement rapide d'une API
- Son évolution
- Son extensibilité

De plus, cela nous permet principalement de gérer les interactions entre la maquette et les données que nous utilisons.

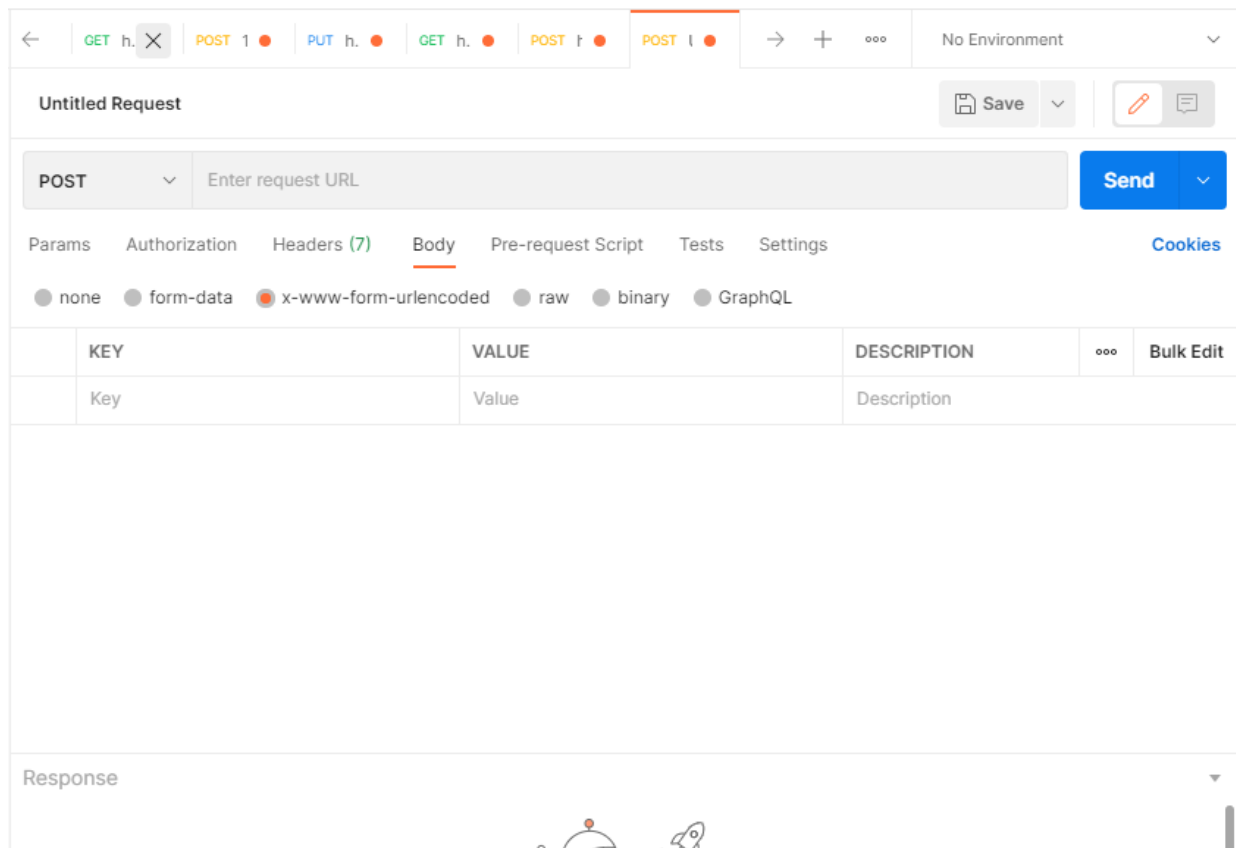
Pour en expliquer le fonctionnement, j'ai dû créer une application sur un serveur NodeJS permettant d'interroger les serveurs de l'entreprise fournissant les différentes données comme la température, l'humidité ou encore la luminosité.

Les données des capteurs étant présentées sous forme d'API (Application Programming Interface), j'ai dû faire un appel sur leur API pour en récupérer les données.

Une API est un ensemble de contenus mis à disposition par un serveur. Dans ce cas, l'entreprise met ses ressources à disposition et sont accessibles par plusieurs URL's aussi appelés routes comme pour effectuer une recherche sur internet mais ses données sont protégées par des tokens.

Ce qui permet de restreindre le contenu qu'à un certain type d'utilisateurs et en maîtriser l'utilisation et de renforcer la sécurité de la diffusion de ces données.

Voici un exemple d'interaction avec une API client par un outil client de gestion de requêtes Web :



L'outil présenté ci-dessus est PostMan, il s'agit d'un outil utilisé qui permet de construire et d'exécuter des requêtes HTTP, de les stocker dans un historique afin de pouvoir les rejouer, mais surtout de les organiser en Collections.

Cet outil est très utilisé par les développeurs car il permet de tester ses requêtes dans un premier temps avant de les imbriquer dans ses applications.

4^{ème} Mission : QR Code

Cette mission consiste à gérer l'interaction avec les serveurs d'une entreprise ayant comme domaine d'activité la gestion de QR Code pour des équipements. L'objectif étant d'implémenter leur système à notre outil et de rajouter une fonctionnalité pour les clients.

Le but étant qu'à chaque création d'un objet dans notre plateforme un QR code unique soit créé chez notre partenaire de manière automatique, générant un « jumeau numérique » sur notre plateforme.

Pour expliquer un peu plus en détail ce qu'est un objet, il s'agit d'une représentation numérique de ce qui se rapproche au plus d'un équipement ou encore d'un capteur dans la vraie vie et de le retranscrire en format numérique.

Par exemple, nous avons une centrale de traitement d'air chez un client, nous allons donc relever les différentes propriétés qui la constituent et les enregistrer dans un même élément sur notre plateforme. Ces différentes propriétés étant liées à cet élément alors il nous sera possible d'interagir avec l'équipement par son « jumeau numérique » présent dans notre outil.

Voici une capture d'écran d'un « Objet » de notre plateforme :

Thing: SEXTANT_Building_ENERGY 2_Id_1

General Information | Properties and Alerts | Services | Events | Subscriptions | Permissions | Change History | View Relationships

Properties | Alerts | Filter | Choose category

My Properties | Add | Duplicate | Delete | Manage Bindings | Refresh

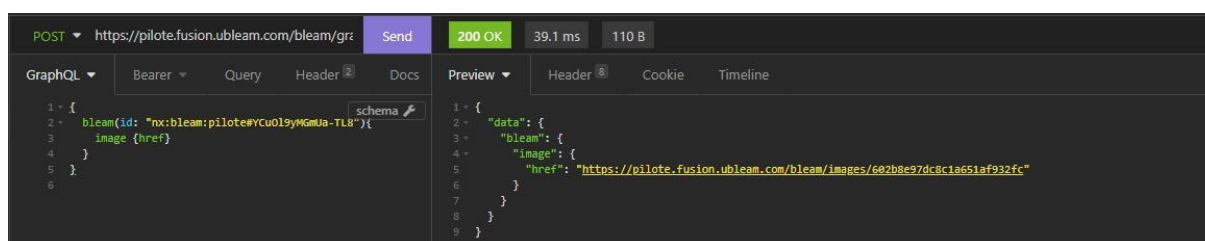
Name	Actions	Source	Default Value	Value	Alerts	Category	Additional Info
No properties							
Inherited Properties							
SEXTANT_Building_Template							
Name	Actions	Source	Default Value	Value	Alerts	Category	Additional Info
-T- Annee_Construction	ⓘ	-	✓	1990	+	0	
-T- Annee_Permis_Construction	ⓘ	-	✓	1990	+	0	
-T- Annee_Renovation	ⓘ	-	✓	2008 - 2018	+	0	
categories_compteurs	ⓘ		✓	categories_compteurs (2)	+	0	SEXTANT_Categorie_compteurs_datashape
-T- Certification	ⓘ	-	✓	BREEAM In Use	+	0	
-T- Classe_Oid	ⓘ	-	✓	BUREAUX	+	0	
compteurs	ⓘ		✓	compteurs (15)	+	0	SEXTANT_DataShape_Compleur_list_template
compteurstemp	ⓘ		✓	compteurstemp (15)	+	0	SEXTANT_DataShape_Compleur_list_template
Conso_energie_list	ⓘ		✓	SEXTANT_Conso_Energetique...	+	0	SEXTANT_Conso_Energetique_List_Template
Contact	ⓘ		✓	Contact (3)	+	0	SEXTANT_Contact_DataShape
Contacts	ⓘ		✓	Contacts (3)	+	0	SEXTANT_Contacts_DataShape_v2
Data_From_Force_GTB	ⓘ		✓	SEXTANT_ENERGY2_data_fr...	+	0	
Data_From_Force_Scan	ⓘ		✓	SEXTANT_ENERGY2_data_fr...	+	0	
-T- Dataroom_Url	ⓘ		✓	https://nextcloud.nextim.fr/index.php/...	+	0	
-T- Decret_Tertiaire_Annee_Reference	ⓘ	-	✓	-	+	0	

Dans cet exemple il s'agit d'un Bâtiment d'un de nos clients, comme vous pouvez le voir cet objet référence les propriétés du bâtiment et est relié aussi à d'autres entités qui ont elles aussi des attributs plus complexes.

La mission a été totalement terminée et m'a permis de m'améliorer dans la gestion d'API GraphQL qui était pour moi une technologie complètement inconnue. J'ai pu de ce fait comprendre en intégralité son fonctionnement.

Le résultat final de l'application pour l'entreprise a pour but de donner la possibilité de pouvoir créer, modifier et mettre à jour un équipement côté « Sextant » et qu'il soit redondant sur notre plateforme ainsi que sur la plateforme fusion. Cette dernière nous permet de créer les QR Code et de pouvoir les utiliser sur notre plateforme.

Voici un Screenshot de l'outil GraphQL que j'ai été amené à utiliser :



Vous pouvez voir ci-dessous un exemple de la représentation d'un QR Code. Il peut être scanné et visualisé dans une interface graphique contenant les informations associées.

Par exemple, les détails d'un équipement ou même d'un bâtiment peuvent être affichés.



5^{ème} Mission : Map Maquette

Cette mission avait pour but de récupérer la maquette 3D d'un client comme dans le cas de la mission API Température mais au lieu d'avoir la maquette sur un fond transparent, j'ai pu intégrer une Map Google en fond.

Il s'agit d'une mission toujours en cours mais voici un exemple du résultat actuel :



Pour cette solution j'ai été amené à utiliser diverses technologies comme :

- BingMaps
- Forge Autodesk
- Javascript/HTML/CSS
- JQuery

J'ai rencontré quelques difficultés sur ce projet car l'implémentation de notre maquette au système de gestion de carte de Bing a été assez complexe à mettre en place du fait de l'intégration de certains critères tels que le passage de nos serveurs en HTTPS pour pouvoir utiliser la géolocalisation et déterminer l'emplacement des bâtiments.

Certaines alternatives ont néanmoins été mises en place afin de pallier ce problème comme on peut le voir en haut à gauche du Screenshot ci-dessus.

J'ai été amené à rajouter des petits sliders afin de faire bouger notre maquette à la main sur le plan 2D pour pouvoir la positionner le temps de pouvoir récupérer des coordonnées plus précises.

Cette mission a été le développement d'une nouvelle fonctionnalité pour l'entreprise comme avec ma mission d'API Température.

En effet, il s'agissait d'un projet non-existant dans l'entreprise mais relevant d'un besoin client. Cela nous a donc permis de créer une plus-value pour les clients.

De plus, cette solution permet aux clients une meilleure visualisation de leurs bâtiments ainsi qu'une représentation géographique.

Néanmoins cette fonctionnalité reste en développement car le besoin clients n'est pas totalement rempli, il reste encore quelques petits réglages à faire.

Cette mission m'a permis de découvrir comment gérer et mettre en place une interaction avec un module de gestion de carte comme « Bings Maps » de Microsoft ainsi que la gestion de ce module pour le rendre compatible avec un besoin client.

Analyse de la situation

Actuellement, et afin de maintenir l'évolution de notre plateforme de « smart-building », l'entreprise Nextiim a besoin de développeurs. Avant que j'intègre l'entreprise il n'y avait que Julien Cousin mon tuteur et un ingénieur réseau pour faire évoluer notre plateforme de gestion de bâtiments « smart-building ».

C'est pourquoi Nextiim a décidé de recruter deux alternants développeurs, dont moi, afin de nous familiariser avec notre plateforme, de pouvoir maintenir son évolution, et de devenir complètement indépendants quant à son évolution dans le temps.

J'ai eu une phase de compréhension et d'apprentissage des différents outils avant de « plonger » directement dans le vif du sujet.

L'outil « ThingWorx » est un outil très complexe à comprendre et à prendre en main mais il s'agit du cœur de notre plateforme. Il a donc fallu que je le maîtrise dans son intégralité afin de pouvoir être opérationnel et efficace au plus vite.

Ceci a pris l'équivalent de deux mois d'adaptation avant d'acquérir les bons mécanismes pour interagir et développer de nouvelles fonctionnalités aux contenus déjà existants.

Je me suis également auto-formé sur certaines technologies que je n'avais jusqu'à présent jamais étudiées. Ce travail d'autoformation est une suite logique au métier de Developpeur web car notre métier est en perpétuel évolution et maîtriser les dernières technologies et celles les plus importantes sur le marché est un atout primordial pour valoriser sa montée en compétence et son efficacité au sein d'une entreprise.

Ce qui a été assez complexe aussi dans mon autoformation a été le fait que l'intégralité du support de « ThingWorx » soit en Anglais et qu'il ne soit pas très actif. Ceci a donc été pour certaines notions assez difficile à comprendre.

Afin de répondre à ces diverses problématiques, je me suis mis à niveau tout au long de mon alternance et j'ai été amené à demander conseil à mon tuteur.

Malgré ces différents problèmes nous avons toujours trouvé des solutions quant à leurs résolutions.

De plus lors de mon arrivée, j'ai été amené directement à découvrir « ThingWorx » et son architecture pour ensuite l'améliorer mais la gestion des données clients et interface avait été pensée de manière statique.

De ce fait il prenait une place considérable sur les serveurs, il a fallu trouver une solution : redévelopper notre plateforme de manière dynamique.

Les différents projets que j'ai réalisés m'ont demandé un temps plus conséquent du fait de l'apprentissage des certains langages et outils informatiques qui m'étaient jusqu'à présent inconnus.

En effet, il y a eu dans certains cas quelques écarts de prévisions de temps de projets du fait de quelques contretemps comme l'incendie du Datacenter d'OVH à Strasbourg. En effet nous avions plusieurs de nos projets hébergés dans ce Datacenter et avons perdu beaucoup de données et de temps de travail que nous avons dû rattraper tout au cours des mois qui ont suivis.

Nous avons également mis en place des règles de travail sur nos serveurs, par exemple séparer les espaces pour les clients et ceux réservés au développement de fonctionnalités. De plus, les serveurs étant un peu désordonnés, nous avons mis en place un ensemble de règles et de normes afin de retrouver une architecture logique et optimisée pour un meilleur avancement de notre environnement de travail.

Nous évoquerons dans la prochaine partie par quels procédés et moyens nous avons dû mettre en place des solutions aux problèmes rencontrés.

Mise en place de solutions et actions

Après avoir pris en main les différents outils dont dispose « Nextiim » j'ai pu réaliser les missions qui m'ont été confiées. Malgré les problèmes évoqués précédemment, des solutions ont été mises en place afin de pallier ces problèmes comme des sauvegardes continues de nos projets afin d'éviter toutes pertes définitives qui auraient pu avoir de très lourdes conséquences pour l'entreprise.

Comme expliqué précédemment la plus grosse crainte pour une entreprise est la sécurité de ses données et projets. C'est pourquoi nous avons trouvé la solution de louer des serveurs dans plusieurs Datacenter à travers la France et non seulement à Strasbourg pour faire face à l'incendie d'OVH Strasbourg auquel nous n'étions initialement pas prêts.

Cette action nous permet d'assurer une certaine pérennité de nos données et d'avoir repensé certaines des fonctionnalités déployées sur nos serveurs et la mise à niveau de nos machines.

Néanmoins le risque zéro n'existe pas mais ce genre de préconisations nous permet de réduire au maximum la future potentielle perte de données indépendante de notre volonté.

Quant aux solutions mises en place par rapport à l'intégration des projets, nous avons actuellement trois serveurs de travail :

- Le serveur de production
- Le serveur de préprod
- Le serveur de développement

Ces serveurs existaient au préalable mais ont été détruits à la suite de l'incendie de Strasbourg, c'est pour cela que nous les avons redéployés et profité pour installer les nouvelles mises à jour de nos logiciels et outils permettant une meilleure sécurité.

Nous avons mis en place des intégrations continues c'est-à-dire que nous développons et testons dans un premier temps nos projets sur le serveur de développement qui est une copie conforme au niveau de l'architecture du serveur de production.

Ceci nous permet de nous retrouver dans la même situation que le serveur de production qui est le serveur dédié aux clients.

Nous avons en plus de ces 3 serveurs déployés d'autres serveurs permettant la gestion de projets comme :

- Le serveur Forge
- Le serveur Matterport

Il s'agit de projets déployés sur de nouvelles machines. Nous effectuons des sauvegardes régulières afin de réduire la perte de données et de nos outils pour éviter de reproduire des situations à risques.

Nous mettons en production dès lors que tous les tests sont validés pour éviter tout bugs sur les données clients. Pour cela nous utilisons des données répliquées des clients mais dans un espace où le client ne peut accéder ; ce qui nous permet de faire des tests qui se rapprochent au mieux de la réalité.

Néanmoins nous avons un serveur de préprod pour faire nos tests juste avant de mettre en production.

Comme évoqué dans l'Analyse de la situation, à mon arrivée la structure de notre solution était orientée et spécifique à chaque client ce qui représentait une duplication des mashups « vues » de notre solution pour chaque client.

Il nous a fallu rendre notre solution dynamique et adaptative à nos clients et mettre en place un principe de permissions afin de leur proposer un accès à différentes fonctionnalités de notre plateforme.

Nous avons donc créé nos « vues » adaptatives à nos clients avec l'utilisation de leurs données afin de pouvoir rendre la plateforme propre à chaque client en fonction de ses besoins.

Ce serveur est souvent utilisé tout comme le serveur de tests/développements, pour la création de fonctionnalités mais plus dédié aux tests.

Pour donner suite à nos pertes de données chez OVH nous avons aussi déployé de nouveaux serveurs de gestion de projets répartis dans toute la France afin de ne plus rencontrer ce type de problèmes.

Analyse des résultats

Les différentes missions que j'ai pu effectuer chez « Nextiim » ont permis une amélioration de la plateforme par l'ajout et l'optimisation de fonctionnalités permettant aux clients un plus grand panel de possibilités.

La plupart des missions que j'ai réalisées ont été intégrées à notre plateforme « Sextant » et pour les autres elles sont en cours d'intégration car nous avons décidé d'intégrer les fonctionnalités les plus importantes et les plus demandées par les clients en priorité car nous adaptons nos intégrations en fonction des besoins clients.

Après avoir mis en place un ensemble de solutions et de plans d'actions comme des normes sur la syntaxe et l'architecture de nos projets au sein de notre solution nous avons pu constater une meilleure efficacité sur nos projets quant à l'organisation.

Cette mise en place a pu montrer son efficacité car nous avons eu des stagiaires durant une période de 10 semaines et il a été plus simple de les intégrer à notre solution et de leur expliquer notre fonctionnement et comment se servir de l'outil en respectant certaines normes et syntaxes afin de mener à bien les projets qui leurs ont été confiés.

L'ajout de fonctionnalités permet donc aux clients d'avoir à disposition un outil plus complet pour leur analyse de bâtiments et permet pour l'entreprise un atout majeur quant à la concurrence et le panel de fonctionnalités disponibles.

De plus le fait d'avoir mis en place des règles et normes dans nos projets a permis d'optimiser le temps de travail et son organisation ce qui permet d'uniformiser les projets et de les rendre compréhensibles par l'ensemble de l'équipe de développeurs qui travaillent sur les différents projets.

La mise en place du dynamisme dans notre solution a permis un énorme gain de performance et de stockage de données pour nos serveurs et une optimisation de l'évolutivité conséquente car nous avons réduit le nombre d'objets présents sur notre plateforme tout en ayant la même utilisation.

C'est pourquoi les résultats actuels me confortent dans l'idée que j'ai pu apporter ma pierre à l'édifice dans l'amélioration de notre plateforme et son évolutivité grâce aux différentes solutions apportées.

Conclusion

A travers mes différentes missions j'ai été amené à maintenir l'évolution de notre plateforme en implémentant de nouvelles fonctionnalités et en améliorant aussi certaines de ses possibilités.

L'intégration de mes missions à l'entreprise a permis une grosse évolution sur notre outil de gestion de bâtiments « smart-building » en partie car notre équipe de développeurs se constitue de 3 personnes : mon tuteur, un alternant et moi-même.

Mon tuteur étant polyvalent et amené à switcher entre les trois entreprises j'ai été amené à devenir très vite autonome dans mes missions, ce qui m'a permis une montée en compétences plus rapide et une complète gestion de mon temps pour mes missions afin d'être plus efficace.

L'évolution de notre plateforme a donc vu une progression fulgurante en l'espace de quelques mois par les fonctionnalités qui ont été développées en plus et les fonctionnalités optimisées permettant un meilleur fonctionnement de notre outil.

Le maintien et l'évolution de « Sextant » a pu être amélioré par les différents processus que nous avons mis en place tel que :

- La séparation des serveurs de développement et de production
- L'implémentation de normes de syntaxe pour que tous les développeurs présents sur l'outil et les futurs développeurs amenés à utiliser l'outil ne soit pas perdus et puissent prendre en main rapidement notre outil.

Mes missions ont pu apporter à l'entreprise un meilleur maintien de la solution quant aux développements de futures fonctionnalités et projets par les normes et règles mises en place.

C'est pourquoi par mes compétences et mes analyses j'ai pu amener un point de vue extérieur qui a permis à notre entreprise une meilleure structure et gestion des projets quant au maintien de l'évolutivité de notre solution.

Ceci nous amène à la question suivante, Est-ce que notre plateforme de « smart-building » pourrait être encore mieux maintenue et optimisée si notre solution était entièrement propriétaire à notre entreprise ?

Références

<https://consolab.fr/wp-content/uploads/2019/08/avis-application-dyson-ventilateur-purificateur.jpg>

<https://www.nextiim.fr/>

<https://fr.wikipedia.org/wiki/Immotique>

<https://www.ptc.com/fr/products/thingworx>

<https://www.dyson.fr/fr>

<https://www.abcclim.net/degres-jour-dju.html>

<https://fr.wikipedia.org/wiki/Node.js>

<https://www.redhat.com/fr/topics/api/what-are-application-programming-interfaces>

https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade

<https://fr.wikipedia.org/wiki/Domotique>

<https://experiences.microsoft.fr/articles/intelligence-artificielle/smart-building/>

https://fr.wikipedia.org/wiki/Hypertext_Transfer_Protocol

<https://forge.autodesk.com/>

<https://fr.wikipedia.org/wiki/JQuery>

<https://www.4cadgroup.com/fr/ptc-thingworx>

<https://fr.wikipedia.org/wiki/OVHcloud>

<https://fr.wikipedia.org/wiki/Wireshark>

https://fr.wikipedia.org/wiki/Internet_des_objets

https://fr.wikipedia.org/wiki/Suite_des_protocoles_Internet

<https://www.commentcamarche.net/contents/539-tcp-ip>

<https://fr.wikipedia.org/wiki/MQTT>

https://fr.wikipedia.org/wiki/Uniform_Resource_Name

<https://fr.wikipedia.org/wiki/GraphQL>

https://fr.wikipedia.org/wiki/Bing_Cartes

https://fr.wikipedia.org/wiki/HyperText_Transfer_Protocol_Secure