

Développement d'une application de digitalisation Full Stack



Lieu du stage : LATECOERE – Tunis - Tunisie

Dates : 29/08/2022 – 23/12/2022

Service Informatique

Nom du responsable de service IT : M. SELMI

Stage supervisé par : M. Housseem SELMI

LATÉCOÈRE

Remerciements

Je tiens à remercier dans un premier temps, toute l'équipe pédagogique et le service des stages de l'EPF pour leur efficacité dans les démarches administratives.

Je remercie tout particulièrement Houssem Edine SELMI, mon tuteur de stage et manager IT du site de Tunis, pour m'avoir accepté en tant que stagiaire au sein de l'entreprise, pour la confiance qu'il m'a accordée et pour le partage de son expertise au quotidien. Son accompagnement, ses explications précises et pédagogiques m'ont permis de rapidement appréhender l'ensemble du processus de développement d'une application web.

J'adresse également mes remerciements au responsable RH Moez MEJRI, pour sa patience, notamment dans la création de la convention de stage et pour son temps lors de notre entretien téléphonique.

Enfin, je remercie Marwa NOOMENE pour sa bienveillance ainsi que l'ensemble des employés de Latécoère pour tout ce qu'ils ont pu m'apporter au cours de cette expérience très enrichissante.

Développement d'une application de digitalisation Full Stack

Résumé

L'objectif de ce stage était d'optimiser la saisie de formulaires notamment, que sont amenés à remplir les employés de Latécoère, principalement dans le secteur production. Ces formulaires sont pour la plupart issus de visites managériales (qualité, sécurité, etc.) et permettent de mentionner les résultats constatés, jusqu'ici réalisés sur papier. Pour faciliter leur saisie, leur accessibilité et l'exploitation des données obtenues, une digitalisation de ces démarches a été initiée via le développement d'une application web. L'application centralise tous ces aspects mais propose aussi d'autres fonctionnalités utiles pour les managers, comme la génération d'indicateurs de performance, de plannings de visites ou encore un suivi pour la passation entre employés d'un même poste.

Mots-clés

Digitalisation, application web, KPI, Full Stack, backend, frontend, Python, HTML, JavaScript, CSS, PostgreSQL

Abstract

The objective of this internship was to optimize the data entry of the forms that Latécoère employees have to fill out, mainly in the production sector. These forms are mostly issued from managerial visits (quality, safety, etc.) and allow to mention the results observed, until now done on paper. In order to facilitate their entry, accessibility and the use of the data obtained, a digitalization of these procedures was initiated via the development of a web application. The application centralizes all these aspects but also offers other useful functionalities for managers, such as the generation of Key Performance Indicators (KPI), visit schedules or even a follow-up for the handover between employees in a same job.

Keywords

Digitalization, application web, KPI, Full Stack, backend, frontend, Python, HTML, JavaScript, CSS, PostgreSQL

Sommaire

Remerciements.....	1
Introduction	5
Présentation de l'entreprise	6
Fiche identité	6
Chiffres clés	6
Structure d'accueil	7
Solutions ou métiers	7
Secteurs d'activités	8
Valeurs et contribution aux enjeux environnementaux	8
Projet d'étude	9
Contexte et détail du problème	9
Visites managériales – VMS & VMQ.....	9
Gemba Walk	9
5S	10
Passation.....	11
Divisions, Programmes, Produits et Ateliers	11
KPI	12
Objectifs du stage.....	12
Méthodologie adoptée	13
Analyse des besoins.....	13
Conception	14
Développement	17
Tests.....	31
Déploiement	31
Synthèse.....	32
Conclusion.....	33
Conclusion technique.....	33
Conclusion personnelle.....	34
Bibliographie	35
Annexes.....	35

Liste des figures

Figure 1. Commits du projet les plus récents	17
Figure 2. Arbre de construction architecturale du projet	19
Figure 3. Fichiers du dossier "maître"	20
Figure 4. Fichier urls.py du dossier "maître"	20
Figure 5. Modèles BaseModel et SoftDeleteModel	21
Figure 6. Modèles Division et Programme	21
Figure 7. CRUD-R pour les modèles division.....	22
Figure 8. Liste identité des employés	23
Figure 9. Vue de l'onglet Division	24
Figure 10. Vue de l'onglet Division-Programme.....	25
Figure 11. Architecture du dossier templates, app configuration	25
Figure 12. Vue de l'onglet Qualifications.....	26
Figure 13. Plannings VMQ	26
Figure 14. Exemple de planning VMQ	27
Figure 15. KPI pour les VMQ	27
Figure 16. Onglet 5S – Critères	28
Figure 17. Exemple de formulaires 5S	29
Figure 18. Vue de l'onglet 5S	29
Figure 19. Formulaire de passation	30
Figure 20. Onglet Passation	30

Liste des tableaux

Tableau 1 . Chiffres clés	7
---------------------------------	---

Glossaire

Acronyme Détail

KPI	Key Indicator Performance
5S	Technique de management japonaise, démarque qualité (Seiri, Seiton, Seiso, Seiketsu, Shitsuke)
VMQ	Visite Managériale Qualité
VMS	Visite managériale Sécurité
MVT	Modèle-vue-template (Architecture Django)
CRUD	Création (create), lecture (read), mise à jour (update) et la suppression (delete) de données
-R	CRUD-R : R pour restaurer les données dans le cas d'un soft delete

Introduction

L'efficacité en entreprise est un principe primordial pour atteindre les objectifs. Cela passe souvent par une optimisation et une répartition réfléchie du temps que l'on accorde aux missions.

Des contrôles réguliers sont fréquemment réalisés dans le secteur production de Latécoère sous forme de visites managériales par les managers afin de s'assurer que la sécurité règlementaire est appliquée mais aussi que la qualité des produits est irréprochable pour répondre aux demandes d'un secteur très exigeant : l'aéronautique.

Ces démarches engendrent une nécessité de rédaction de rapport ou de saisie de formulaires pour y recenser les résultats obtenus, ces derniers étant réalisés sur papier. Afin d'exploiter ces résultats, une analyse manuelle fastidieuse permettant d'obtenir les KPI requis est nécessaire, administrer les visites est également très chronophage pour générer les plannings et bien plus encore.

Une digitalisation de ces procédures ne pouvait qu'être envisagée pour répondre au besoin de l'entreprise et celui de ses employés pour un gain de temps quotidien et une optimisation poussée de ces démarches était donc plus que judicieuse.

Comment mettre en place les outils informatiques nécessaires pour répondre à ces besoins d'amélioration afin de renforcer et de mettre en sécurité la mission d'efficacité et la quête de performance de l'entreprise ?

L'entreprise et ses secteurs d'activités seront dans un premier temps présenté avant de détailler ses besoins d'augmentation de productivité et de mettre en relation de la solution apportée par ce stage pour répondre à toutes ces attentes.

Pour ce faire, le projet d'étude sera dévoilé en commençant par la méthodologie de conception adoptée. Les étapes de développement de l'application de digitalisation full stack seront ensuite détaillées avant d'évoquer l'utilité des tests de validation.

L'intégralité des démarches du projet d'étude qui ont permis d'aboutir à un produit fini et utilisable par les employés de Latécoère, sur le site Tunisien ou ailleurs sera décrite.

Enfin, le déploiement de l'application et la mise en production seront exposés.

Présentation de l'entreprise

Fiche identité

Latécoère est une société française de fabrication et d'ingénierie aérospatiale. Elle a été fondée en 1917 et son siège social est situé à Toulouse, en France.

L'entreprise a débuté comme fabricant de pièces d'avion et s'est depuis développée pour offrir une large gamme de produits et de services dans l'industrie aérospatiale.

L'entreprise est spécialisée dans la conception, le développement et la fabrication de composants et de systèmes d'aéronefs, ainsi que d'aérostructures et de services d'ingénierie.

La mission de Latécoère est d'être un fournisseur de premier plan de solutions aérospatiales innovantes et fiables qui améliorent les performances, la sécurité et le confort des avions. La vision de l'entreprise est d'être reconnue comme un partenaire de confiance de choix par les clients et les parties prenantes du monde entier.

Parmi les produits et services de Latécoère figurent les portes d'avion, les intérieurs, les systèmes de câblage et les nacelles (le boîtier extérieur d'un moteur d'avion). L'entreprise travaille avec de grands constructeurs aéronautiques tels qu'Airbus, Boeing et Bombardier.

Les valeurs de Latécoère comprennent l'innovation, l'excellence, l'intégrité et le respect des personnes et de l'environnement. Latécoère est notamment reconnu pour ses qualités professionnelles, fiables et innovantes.

Chiffres clés

Latécoère est spécialisé dans la fabrication et la commercialisation de structures pour le secteur aéronautique et produit également des systèmes embarqués et du câblage. Les produits fabriqués sont très variés et peuvent être regroupés par famille dont les ventes se répartissent comme suit :

- Aérostructures (46,2%)
- Equipements et systèmes embarqués (53,8%)

Concernant les aérostructures, il s'agit notamment de sections de fuselage, de portes et d'issues de secours. Latécoère se démarque beaucoup grâce à son savoir-faire dans les équipements et systèmes embarqués car le groupe est numéro 1 en Europe pour les harnais électriques, numéro 2 mondial pour les câblages embarqués, et possède une grande part de marché pour le mobilier avionique, les bancs de test, les équipements et systèmes vidéo embarqués, etc.

Les ventes par marché sont réparties entre les secteurs civil et militaire, chaque part représentant respectivement 97% et 3% du marché.

La répartition géographique des ventes est la suivante :

- France (49,2%)
- Europe (14,5%)
- Amérique (32,9%)
- Asie (2,9%)
- Autres (0,5%)

	2017	2018	2019	2020	2021
Chiffre d'affaires (milliers €)	657 409	659 249	713 098	413 232	375 877
Nombre d'employés	4451	4958	-	-	4764
Nombre d'implantations	-	-	-	-	23

Tableau 1 . Chiffres clés

Structure d'accueil

Latécoère a de nombreux sites de production et de services à travers le monde.

La répartition des sites est de 49% en Europe, avec 13 sites dont 7 en France, 3% en Asie avec 2 sites, 28% en Amérique pour un total de 6 sites et enfin 20% en Afrique avec un site Marocain et un autre Tunisien, ce dernier étant la structure où ce stage s'est déroulé.

Latécoère a un site de production à Tunis qui se spécialise dans la fabrication de pièces d'avion en aluminium et en acier inoxydable. Le site de Tunis est équipé de machines de découpe laser, de pliage et de soudage de haute précision.

Le travail a été effectué au sein du service informatique dans des locaux dédiés. Des bureaux équipés et personnels sont mis à disposition des employés. Ce stage s'est donc déroulé dans de bonnes conditions de travail auprès du tuteur de stage et des collègues.

Solutions ou métiers

Latécoère propose de nombreuses solutions innovantes qui répondent à des besoins spécifiques. Des métiers spécialisés ont été créés pour faire face à ces attentes dans la conception, le développement et la fabrication de composants et de systèmes d'avion, mais aussi dans les aérostructures et les services d'ingénierie.

Les solutions apportées sont notamment les suivantes :

- Portes d'avion
- Intérieurs d'avion
- Systèmes de câblage
- Nacelles (carcasse extérieure des moteurs d'avion)
- Services d'ingénierie (conception, développement, prototypage, essais)

Latécoère travaille avec de grands constructeurs aéronautiques tels qu'Airbus, Boeing et Bombardier, les solutions ou métiers sont donc très variés.

Secteurs d'activités

L'entreprise est active dans les segments de l'aviation commerciale, militaire et de l'espace.

En ce qui concerne l'aviation commerciale, Latécoère propose une gamme complète de produits et de services pour les avions de ligne, tels que des portes de cabine (portes d'embarquement, portes de service, portes de cockpit), intérieurs d'avion (sièges, revêtements de cabine, placards), systèmes de câblage (câblage électrique, câblage de communication, câblage de contrôle) ou encore des nacelles (carcasse extérieure des moteurs d'avion).

Latécoère fournit également des produits et services pour les avions militaires. On retrouve principalement des portes de cockpit, des puits de train d'atterrissage ainsi que des systèmes de câblage encore une fois.

Enfin, l'entreprise ne se limite pas à l'aéronautique en proposant également des solutions pour les satellites et les lanceurs spatiaux. Par exemple, les structures de satellite, les systèmes de propulsion et les équipements de télécommunications sont des produits phares de l'entreprise.

Valeurs et contribution aux enjeux environnementaux

Latécoère a créé une politique de développement durable qui s'aligne sur 11 des 17 objectifs de développement durable des Nations Unies.

Cette politique est organisée autour de trois piliers qui visent à avoir un impact positif sur la société. Cette démarche vient renforcer l'engagement des sites du groupe en faveur de leurs employés et de leurs communautés locales.

Elle offre également de nouvelles perspectives et des moyens pour renforcer la responsabilité sociétale de Latécoère dans son écosystème. Sous l'autorité du Conseil d'administration, une gouvernance opérationnelle sera mise en place à tous les niveaux de l'entreprise pour garantir la réalisation des actions prévues dès 2022.

Projet d'étude

Contexte et détail du problème

Pour atteindre ses objectifs, il est primordial pour une entreprise d'optimiser et de répartir de manière réfléchie le temps qu'elle consacre à ses missions. Cela permet d'améliorer son efficacité.

Visites managériales – VMS & VMQ

Dans le secteur de la production de Latécoère, des contrôles réguliers sont effectués sous forme de visites managériales par les managers afin de s'assurer que la sécurité réglementaire est appliquée et que la qualité des produits est irréprochable, ce qui est essentiel dans un secteur exigeant comme l'aéronautique.

Ces démarches impliquent la rédaction de rapports ou la saisie de formulaires sur papier pour enregistrer les résultats obtenus. Pour exploiter ces résultats, il est nécessaire de réaliser manuellement une analyse fastidieuse pour obtenir les KPI requis ou tout autres statistiques, que ce soit de performance, de récurrence, ou tout autres indicateurs pertinents pour les démarches qualité du secteur.

De plus, avant même de faire ces visites managériales tels que les VMS ou les VMQ, la planification prend beaucoup de temps. En effet, les visites prennent en compte des paramètres bien précis afin que ces dernières soient le plus pertinent possible. Pour ce faire, chaque manager qui possède la qualification correspondante supervise une petite équipe de quelques employés mais n'en visite que deux à la fois au maximum afin d'être le plus précis possible dans les observations.

Un manager visite donc en priorité son équipe à raison de deux personnes en simultané mais pourra contrôler d'autres employés une fois son équipe visitée. Le contrôle qualité et sécurité privilège une équité dans le nombre de visite pour la totalité des employés plutôt qu'un respect constant des équipes pour les visites bien qu'elles soient privilégiées si possible.

Ce schéma se répète à l'échelle de centaines d'employés rien que sur le site Tunisien pour des dizaines de managers associés, de façon hebdomadaire. Une fois l'intégralité des employés visités, la boucle est alors réinitialisée.

Gemba Walk

Le Gemba Walk est une technique de gestion qui vise à améliorer l'efficacité de l'entreprise en identifiant les opportunités d'amélioration. Cela se fait en se rendant sur le lieu de travail (le "gemba" en japonais) et en observant et analysant les activités et les processus du secteur production dans le cas de ce stage mais cette technique peut s'étendre à d'autres secteurs. Le Gemba Walk est souvent utilisé dans le contexte de la philosophie Lean, qui vise à éliminer les gaspillages et à améliorer l'efficacité.

Les managers sont généralement chargés de réaliser le Gemba Walk pour le secteur production de Tunis, pouvant parfois être accompagnés de membres de l'équipe de direction ou de consultants. Ils observent les activités et les processus en détail et posent des questions aux employés pour mieux comprendre comment ils travaillent et quelles sont les difficultés qu'ils rencontrent.

Le Gemba Walk est une méthode qui permet de s'appuyer sur les observations et les suggestions des employés pour développer des solutions pratiques et concrètes aux problèmes et aux opportunités d'amélioration identifiés.

Cette méthode a donc aussi été implémenter dans la solution apportée pour encore une fois rentrer dans cette quête de productivité et d'optimisation.

5S

La technique 5S est une approche de gestion qui vise à améliorer la productivité et l'efficacité en organisant et en optimisant l'espace de travail. Elle se base sur cinq principes :

1. Seiri (tri) : cela consiste à trier et à éliminer tout ce qui n'est pas nécessaire dans l'espace de travail, afin de ne garder que l'essentiel. Cela peut inclure tout type d'éléments, tels que les outils, les documents, les équipements, etc. Le but est de ne garder que ce qui est utile et de libérer de l'espace en éliminant le reste.
2. Seiton (ordre) : cela consiste à mettre chaque chose à sa place, de manière à ce qu'elle soit facilement accessible et identifiable. Cela peut inclure l'utilisation de panneaux d'affichage, de rayonnages, de boîtes ou de tout autre système de rangement qui permette de trouver rapidement ce que l'on cherche.
3. Seiso (propreté) : cela consiste à maintenir l'espace de travail propre et en bon état, afin de minimiser les risques d'accidents et de maximiser l'efficacité. Cela peut inclure le nettoyage régulier du lieu de travail, la maintenance des équipements, etc.
4. Seiketsu (standardisation) : cela consiste à mettre en place des règles et des procédures pour maintenir l'espace de travail organisé et propre. Cela peut inclure la définition de tâches et de responsabilités pour chaque membre de l'équipe, la création de plans de nettoyage, etc.
5. Shitsuke (discipline) : cela consiste à inculquer l'importance de l'organisation et de la propreté dans l'espace de travail et à en faire une habitude quotidienne.

Tout comme le Gemba Walk, la technique 5S est souvent utilisée dans le contexte de la philosophie Lean pour améliorer l'efficacité et la qualité dans l'entreprise et s'associe fréquemment avec la méthode agile.

Dans le cadre de la démarche qualité et performance, la méthode 5S est utilisé dans le secteur production où plusieurs critères préalablement définis doivent être régulièrement validés lors des contrôles d'application. Le rapport écrit de ces contrôles ainsi que les différents critères n'étaient jusqu'ici pas digitalisés, ce qui rendait difficile l'exploitation des résultats.

Passation

Le rythme 5X8 consiste à travailler cinq jours par semaine, pendant huit heures par jour. Cela peut être organisé de différentes manières, par exemple en travaillant du lundi au vendredi ou en travaillant du mardi au samedi avec deux jours de repos consécutifs.

Le rythme 5X8 est souvent utilisé dans le secteur de la production, ce qui est le cas sur le site Latécoère de Tunis car il permet de maintenir une continuité de l'activité tout en offrant deux jours de repos consécutifs à chaque employé.

Un même poste est donc occupé par un employé différent selon les horaires de la journée et le jour de la semaine. C'est pour cela qu'une démarche supplémentaire vient s'ajouter pour simplifier cette procédure : la passation. La passation entre employés d'un même poste est donc le processus par lequel un employé (appelé "relais") prend le relais d'un autre employé (appelé "relégué") sur un même poste de travail.

La passation nécessite donc de compléter un formulaire indiquant les spécificités du travail en cours, d'éventuelles remarques ou toute autres informations qui serait utile au "relais". Cela permet donc de prévoir des possibles périodes de co-présence si nécessaire mais est généralement auto-suffisant et fait office de transition. Ces formulaires seront donc digitalisés via la solution apportée pour centraliser ces procédures.

Divisions, Programmes, Produits et Ateliers

Dans le simple but de simplifier la compréhension future du projet d'étude, les termes division, programme, produit et ateliers seront parfois désignés en anglais (division, program, product et workshop) car la solution apportée a été développée en anglais vu que cette dernière sera déployée à l'internationale, ne se limitant pas aux sites francophones de Tunis ou encore de Toulouse par exemple.

La division désigne le site Latécoère sur lequel sont enregistrés les programs, workshops et items. Dans le cadre de ce stage, le site considéré est celui où le stage s'est déroulé, c'est-à-dire à Tunis.

Un programme dans une entreprise peut être un ensemble de projets ou d'initiatives qui visent à atteindre un objectif commun. Par exemple, un programme de développement de nouveaux produits peut inclure plusieurs projets de recherche et développement visant à lancer de nouveaux produits sur le marché. Dans le secteur production, il s'agit d'un assemblage parent comme un A320 par exemple.

Les produits sont un bien matériel proposé à la vente. Il peut être manufacturé ou issu de l'achat de matières premières et transformé par l'entreprise avant d'être mis sur le marché. Un produit peut désigner un élément ou un produit individuel.

Un atelier est un ensemble de travaux qui a pour objectif de discuter et de travailler sur un sujet précis. Chez Latécoère, un atelier est donc un espace de travail dédié à la production ou au travail

manuel. Par exemple, dans le secteur fabrication, l'atelier peut être l'espace où sont assemblés les différents composants des items ou encore désigné une référence qui permettra aux employés d'établir une structure dans les produits.

Chaque site est spécialisé dans des programmes et produits spécifiques et possède donc des ateliers différents adaptés. La nécessité pour l'entreprise d'avoir une vision globale sur tout ces éléments est donc fondamentale pour garantir un suivi et un accès optimisé ainsi que de permettre des modifications simplifiées dans cette hiérarchie de structure d'assemblage, chacun dépendant d'un site particulier.

KPI

Les Key Performance Indicators (KPI) sont des indicateurs de performance clés qui permettent d'évaluer l'efficacité et le succès d'une entreprise. Ils sont utilisés pour suivre l'évolution de l'entreprise et identifier les opportunités d'amélioration.

Il existe de nombreux types de KPI qui peuvent être utilisés pour mesurer la performance d'une entreprise, tels que les KPI financiers (par exemple, le chiffre d'affaires, le bénéfice net, le retour sur investissement), les KPI de ventes (par exemple, le taux de conversion, le panier moyen, le taux de fidélisation) et les KPI de marketing (par exemple, le taux de clic, le taux de rebond, le nombre de leads générés).

Le stage s'est déroulé dans un contexte d'optimisation des démarches dans le secteur fabrication dans ce sont les KPI de production qui sont ont été le centre d'intérêt. Les principaux concernés sont par exemple les indicateurs de productivité de la qualité de la fiabilité ou encore de sécurité.

Il est important de sélectionner les KPI pertinents pour l'entreprise et qui reflètent ses objectifs et sa stratégie. Il est également essentiel de suivre régulièrement ces indicateurs pour s'assurer que l'entreprise évolue dans la bonne direction et pour identifier les opportunités d'amélioration. La génération de ces KPI a donc été gardée à l'esprit dans toutes les étapes de conception de la solution apportée afin de garantir la pertinence de ces derniers.

Objectifs du stage

Cette grande quantité d'informations représente une base de données très pertinente pour générer des indicateurs essentiels au bon développement de l'entreprise mais aussi un temps considérable à exploiter lorsque ces informations sont saisies sur papier.

On comprend donc rapidement les enjeux de ce manque d'efficacité au quotidien et ce stage devait répondre à ce problème en proposant une solution de digitalisation, accessible par tous selon les droits d'accès, efficace et ergonomique pour une prise en main rapide et sans formation.

La digitalisation de ces procédures permet donc de répondre aux besoins de l'entreprise et de ses employés en leur faisant gagner du temps quotidien et en optimisant ces démarches de manière poussée.

Méthodologie adoptée

Analyse des besoins

Dans un premier temps, le tuteur de stage a participé à des réunions de projet pour comprendre les enjeux et les objectifs de l'application et a été en relation étroite avec les utilisateurs pour comprendre leurs attentes. Un cahier des charges répertoriant le besoin des utilisateurs a donc été établi.

Une étude a ensuite été réalisée pour connaître les outils existants et déterminer ceux qui étaient le plus adaptés pour la conception de la solution qui sera apportée.

Solutions envisageables

Plusieurs solutions auraient pu être envisagées pour répondre à ces besoins tels que :

- La mise en place d'une gestion électronique de documents (GED) : il s'agit d'un logiciel qui permet de stocker, de gérer et de partager des documents électroniques de manière centralisée. La GED permet de réduire les coûts liés à l'impression et à la gestion de l'espace de stockage, et facilite la collaboration entre les différents membres de l'équipe.
- OCR (Optical Character Recognition) : il s'agit d'un logiciel qui permet de numériser des documents papier et de les transformer en fichiers électroniques modifiables. L'OCR permet de faciliter la recherche de documents et de les exploiter de manière plus efficace.
- Workflow : il s'agit d'un logiciel qui permet de définir et de suivre un processus de travail, en automatisant certaines tâches et en permettant de suivre l'avancement du travail. Le workflow permet d'optimiser la gestion de projet et de réduire les temps de traitement.
- Cloud : il s'agit d'une solution de stockage en ligne qui permet de stocker et de partager des documents de manière sécurisée. La solution cloud permet de réduire les coûts liés au stockage et de faciliter l'accès aux documents de l'ensemble de l'équipe.
- Applications web : il s'agit de développer une application web pour faciliter l'accès à des informations ou à des processus en entreprise. Par exemple, une application peut permettre aux employés de remplir des formulaires ou de consulter des données en temps réel.

Solution retenue

En considérant les besoins des utilisateurs mais aussi ceux de l'entreprise d'un point de vue sécurité, confidentialité et accessibilité, le choix s'est porté sur une digitalisation intelligente et optimisée en retenant comme solution au problème le développement d'une application web car cet outil répondra de manière plus précise aux attentes de toutes les parties. Ce choix est celui qui présente

le plus d'avantages car une application est très évolutive si la conception est flexible et c'est un outil qui pourra être mis à jour selon l'évolution des besoins.

L'objectif du stage a donc été de développer une application web de digitalisation de l'ensemble des démarches et procédures précédemment énoncées pour faciliter les visites managériales, la génération des plannings, le suivi de la démarche qualité tels que le Gemba Walk ou encore le 5S, la gestion des produits, programmes et ateliers selon le site de production, tout cela en un même outil informatique, accessible pour tous et facile à prendre en main.

L'intégralité du projet a été réalisé de manière individuelle et le projet a été abouti dans le temps imparti : quelques jours avant la fin du stage. Le tuteur de stage a toujours été en étroite relation avec le stagiaire avec des réunions de suivi quotidiennes notamment.

Conception

Choix des technologies utilisées

Le backend en Python est utilisé pour développer les parties d'une application web qui ne sont pas visibles pour l'utilisateur final. Il s'occupe de toutes les tâches qui se passent en arrière-plan, comme l'accès à la base de données, le traitement des requêtes HTTP et la génération de la réponse à envoyer au client.

Le backend en Python est particulièrement apprécié pour sa flexibilité et sa rapidité de développement. De nombreux frameworks, comme Django ou Flask, facilitent la création de backend en Python en offrant des outils pratiques pour la gestion des routes, la gestion des bases de données et la gestion des utilisateurs.

En utilisant un backend en Python, les développeurs peuvent créer rapidement des applications web performantes et évolutives, qui répondent aux besoins de l'entreprise et de ses utilisateurs.

Backend – Python & Framework Django

Django est un framework de développement web open source en Python. Il a été conçu pour rendre le développement de sites web rapide et facile. Django s'appuie sur le principe du "don't repeat yourself" (DRY) et met en avant la modularité et l'extensibilité.

Avec Django, il est possible de développer rapidement des applications web grâce à sa bibliothèque de composants prédéfinis et à ses outils de gestion de base de données. Django offre également de nombreuses fonctionnalités de sécurité, telles que la protection contre les attaques par injection de code et les attaques par force brute.

En plus de ces fonctionnalités, Django permet de gérer facilement les utilisateurs, les groupes et les permissions, ainsi que de mettre en place un système de gestion de contenu (CMS). Il est également possible de développer des API REST avec Django grâce à sa bibliothèque Django Rest Framework.

Un backend en Python a donc été développé en travaillant avec le framework Django car ce dernier présente de nombreux avantages, notamment en matière de sécurité et de performances évolutives et ses outils d'intégration avec les autres langages sont très pratiques, sa liaison avec le frontend en est simplifiée. Sa prise en main est relativement rapide si l'on intègre facilement son principe de fonctionnement.

Base de données : PostgreSQL

PostgreSQL est un système de gestion de bases de données relationnelles (SGBDR) open source. Il est utilisé pour stocker et gérer de grandes quantités de données structurées de manière efficace et sécurisée. PostgreSQL est souvent utilisé pour le développement de logiciels, le déploiement d'applications web, la gestion de bases de données pour les entreprises et les organisations, et pour le stockage de données de grande envergure, ce qui fait de PostgreSQL un système parfaitement adapté aux besoins de l'application.

En effet, PostgreSQL présente de nombreux avantages, notamment en matière de :

- **Flexibilité** : PostgreSQL permet de stocker des données de n'importe quel type, y compris des types de données personnalisés. Il offre également de nombreuses options de configuration pour adapter la base de données aux besoins de l'application.
- **Performances** : PostgreSQL est connu pour être rapide et fiable, ce qui en fait un choix populaire pour les applications à fort trafic.
- **Sécurité** : PostgreSQL propose de nombreuses options de sécurité, comme l'authentification et la gestion des droits d'accès, pour protéger les données sensibles.
- **Interopérabilité** : PostgreSQL est compatible avec de nombreux autres systèmes et peut être intégré facilement à de nombreuses applications grâce à son API.
- **Communauté active** : PostgreSQL est soutenu par une large communauté d'utilisateurs et de développeurs qui contribuent activement au développement de l'outil et offrent de l'assistance en cas de besoin.

Pour se connecter depuis le backend à une base de données PostgreSQL, il suffit d'avoir un client de base de données, puis s'y connecter en fournissant les informations suivantes :

- Adresse du serveur de base de données (par exemple, localhost ou l'adresse IP du serveur)
- Nom de l'utilisateur et mot de passe pour accéder à la base de données
- Nom de la base de données à laquelle on souhaite se connecter

Django permet de s'y connecter pour y accéder en lecture et en écriture et ainsi d'y être lié dans le backend en renseignant les informations de connexions dans les paramètres du framework.

Frontend – HTML, CSS & JavaScript

Le frontend, ou côté client, est la partie d'une application web qui est visible par l'utilisateur et qui s'exécute dans son navigateur web. Le frontend est généralement construit à l'aide de technologies telles que HTML (HyperText Markup Language), CSS (Cascading Style Sheets) et JavaScript.

HTML est un langage de balisage qui permet de structurer le contenu d'une page web en utilisant des balises spécifiques. CSS est un langage de style qui permet de décrire l'apparence d'un document HTML en définissant des règles de mise en forme pour les différents éléments de la page.

JavaScript est un langage de programmation qui permet de rendre une page web interactive en ajoutant du comportement et des fonctionnalités à l'aide de scripts exécutés côté client.

L'association de ces trois langages ont permis de générer un frontend efficace qui joue un rôle important dans l'expérience utilisateur d'une application web en fournissant une interface visuelle conviviale et interactive.

Django propose des outils permettant d'accéder à des variables Python dans un fichier HTML afin de pouvoir les lire ou les manipuler, ce qui permet une liaison fluide et simplifiée en plus d'une communication du frontend avec le backend, ou côté serveur, en envoyant et en recevant des données via des requêtes HTTP.

Une fois les technologies choisies et la structure mise en place, il a fallu se pencher sur la conception de l'interface utilisateur en imaginant un design intuitif et simple d'utilisation. Par la suite, les fonctionnalités de l'application, la création du modèle de données et de l'architecture logicielle de l'application ont été définies et mis en place.

Les étapes de conception étant terminées, le développement de l'application peut commencer.

Développement

Contrôle de version Git

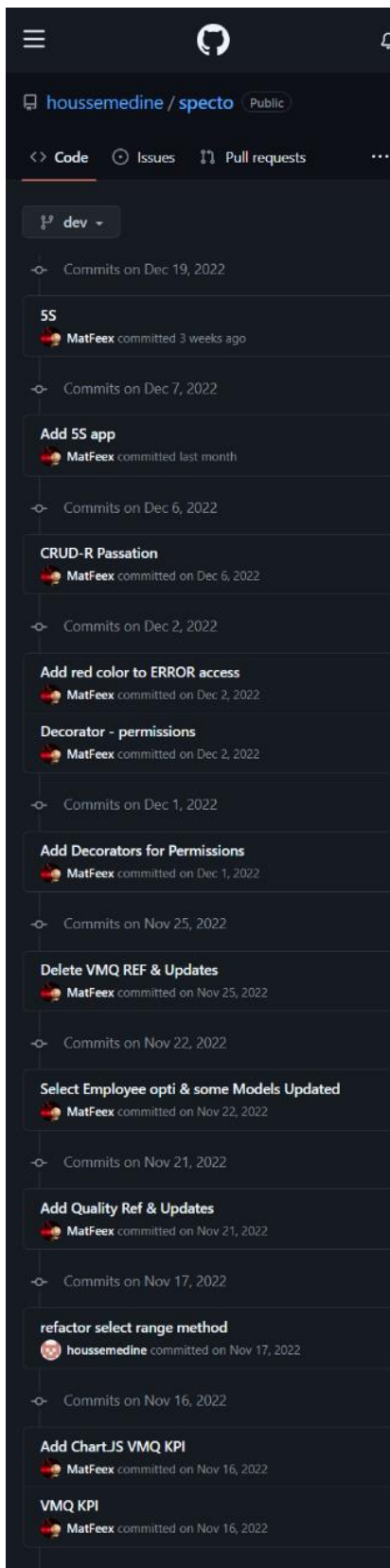


Figure 1. Commits du projet les plus récents

Git est un logiciel de contrôle de version qui permet de suivre les modifications apportées à un projet de manière organisée et efficace. Il est principalement utilisé dans le développement de logiciels, mais peut être utilisé pour suivre les modifications apportées à n'importe quel type de fichier.

Lorsqu'un projet est versionné avec Git, chaque modification apportée au projet est enregistrée de manière séquentielle dans un historique. Cet historique permet de retracer l'ensemble des modifications apportées au projet et de revenir à une version précédente si nécessaire. Git offre également la possibilité de travailler de manière collaborative sur un projet en permettant à plusieurs personnes de contribuer en même temps.

Chaque personne peut travailler sur sa propre copie du projet (appelée "branche") et fusionner ses modifications avec le projet principal lorsqu'elles sont prêtes. Cela permet de travailler de manière efficace sur un projet en équipe et de gérer les conflits de modification de manière structurée.

Pour utiliser les commandes de git, il faut tout d'abord l'installer sur l'ordinateur de travail. Ensuite, un dépôt (repository en anglais) a été créé afin de contenir le projet. Le répertoire git a la même structure qu'un répertoire local sur un PC donc rien de nouveau de ce côté-là.

L'ajout des fichiers au dépôt se fait en utilisant la commande "git add <argument>" et l'enregistrement de ces modifications avec la commande "git commit -m <message>". Les commits les plus récents sont visibles sur la capture d'écran ci-contre et chacun d'eux retrace avec précision la version de la date du commit.

Il suffit de saisir la commande "git push origin <source branch : target branch>" pour valider et envoyer les éléments sur le répertoire git. Il est également possible de créer des branches pour travailler sur des fonctionnalités distinctes et de les fusionner plus tard dans la branche principale (appelée "master"). Git offre nombreuses autres commandes et fonctionnalités avancées qui ont parfois été utilisées.

Ce répertoire a été nommé après une décision commune “**Specto**” et ce sera le nom de l’application. **Specto** peut faire référence à la variété des solutions apportées à l’image du nombre de couleur d’un spectre de lumière.

Ce stage ayant été réalisé de manière individuelle, une seule branche de développement a été créée, nommée “ dev ” et a permis de faire le lien avec le tuteur de stage pour qu’il puisse tester le projet ou, plus tard, mettre l’application en production.

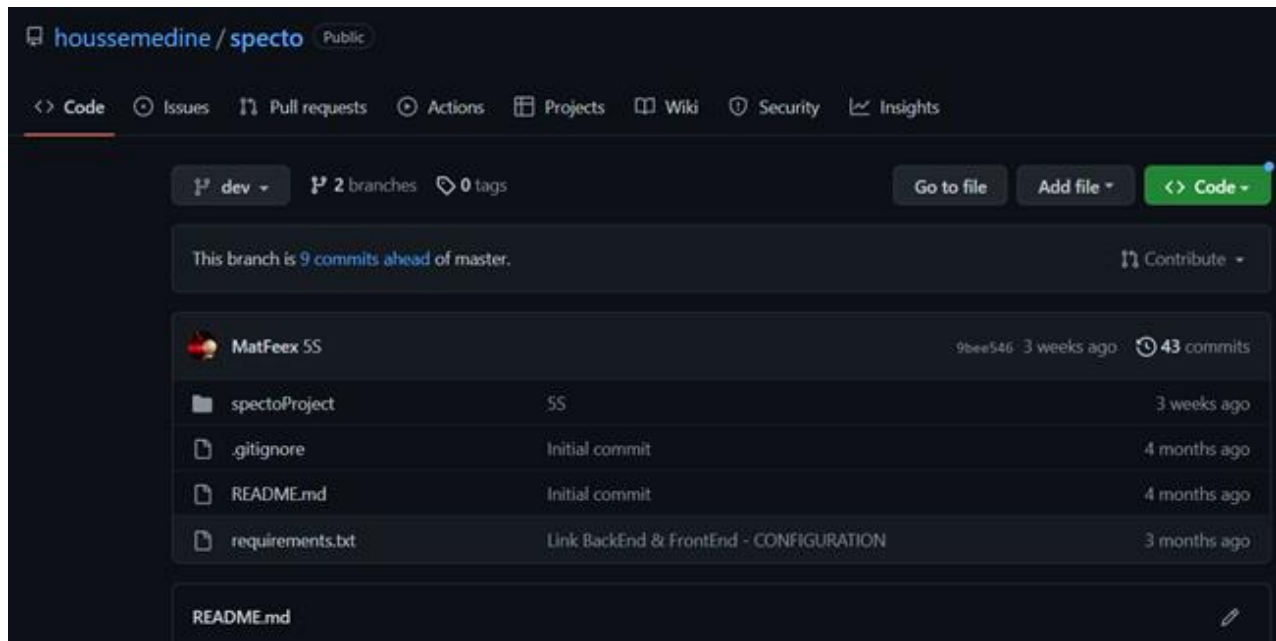


Figure 1. Répertoire git du projet sur le compte GitHub du tuteur de stage

Git a donc permis de sécuriser le travail sur le cloud et de faciliter la gestion du projet car son utilisation propose des fonctionnalités très complètes et a répondu aux besoins de développement.

Backend Python

Django permet aux développeurs de se concentrer sur la création de leur application sans se perdre dans les détails techniques.

Architecture

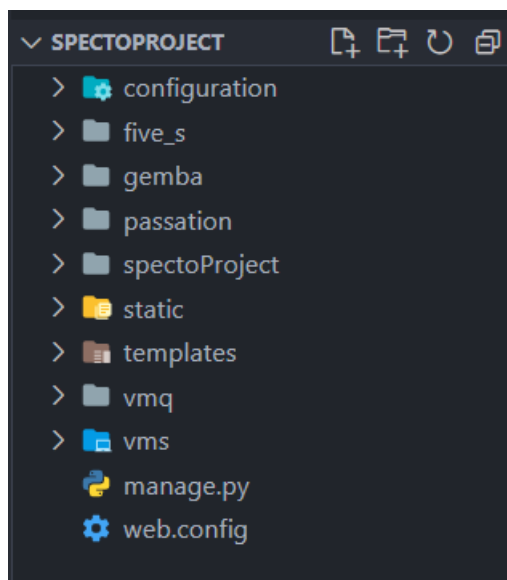
L'architecture de Django est basée sur le modèle-vue-template (MVT). Cela signifie que les données sont gérées par un modèle, l'interface utilisateur est gérée par une vue et le design est géré par un template.

Le modèle est responsable de la gestion des données de l'application. Il utilise une base de données pour stocker les données et fournit des outils pour manipuler ces données, comme la création, la lecture, la mise à jour et la suppression de données (CRUD).

La vue est responsable de la gestion de l'interface utilisateur de l'application. Elle reçoit les requêtes HTTP de l'utilisateur et renvoie une réponse HTTP, généralement sous forme de template HTML. La vue peut être implémentée sous forme de fonction ou de classe.

Le template est responsable de la présentation des données à l'utilisateur. Il contient du code HTML qui peut être personnalisé avec des variables et des balises spéciales fournies par Django. Le template est rendu par la vue et envoyé à l'utilisateur sous forme de réponse HTTP.

L'architecture MVT de Django permet de séparer clairement les différentes couches de l'application, ce qui facilite le développement et la maintenance. Cela permet également de changer facilement l'apparence de l'application en modifiant simplement le template, sans avoir à toucher au modèle ou à la vue.



L'architecture Django de l'application web est structurée et Django génère les fichiers de base nécessaires à son fonctionnement en respectant l'arbre de construction conventionnel via une commande prévue à cet effet. Les différents dossiers visibles ci-contre sont pour la plupart des fonctionnalités implémentées qui sont regroupées par application, chacune recensée dans un dossier.

Ce projet possède donc 6 applications : configuration, five_s, gemba, passation, vmq et vms.

Figure 2. Arbre de construction architecturale du projet

Chaque application possède un sous dossier "templates" qui accueillera les fichiers html propre à l'application en question, qui consisteront le frontend. Le dossier "templates" présent dans la racine du projet accueille les fichiers html communs et réutilisés, souvent sous forme de composant html qui pourront être inclus dans n'importe quelle page du frontend.

Une dossier "migrations", généré automatiquement par Django lorsque les modifications sont appliquées par exécution d'une commande, est souvent présent dans chaque application et permet de garder un historique de ces changements, en relation avec la base de données.

Le dossier "static" n'est pas une application. Dans Django, les fichiers "static" sont des fichiers qui ne changent pas souvent et qui sont généralement utilisés pour la mise en forme de l'application (feuilles de style CSS, scripts JavaScript, images, etc.). Ces fichiers sont gérés de manière différente des autres fichiers de l'application et sont généralement placés dans un répertoire nommé "static" à la racine de l'application.

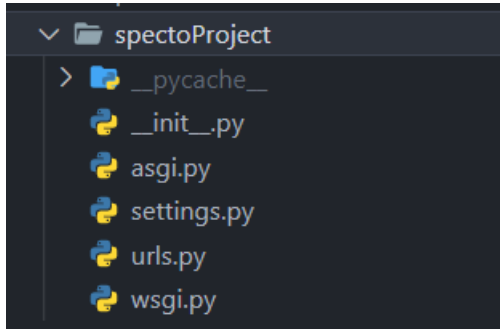


Figure 3. Fichiers du dossier "maître"

Le dossier "spectoProject" n'est autre que le dossier "maître" du backend et recense notamment le fichier settings.py, un fichier de configuration qui contient les paramètres de l'application. Il peut être utilisé pour configurer divers aspects de l'application, tels que la base de données, le système de fichiers, les middlewares, les applications installées et bien plus encore.

Le fichier settings.py est également utilisé pour définir des variables de configuration telles que le nom de l'application, le fuseau horaire, la langue, le format de date et d'heure et bien d'autres encore.

Le fichier wsgi.py (Web Server Gateway Interface) est un fichier utilisé par Django pour configurer l'interface entre l'application et le serveur web. Il contient une variable d'application qui pointe vers l'application Django, qui peut être utilisée par le serveur web pour la déployer. Il sera configuré ultérieurement pour la mise en production.

```
spectoProject > urls.py > ...
1  from django.contrib import admin
2  from django.urls import path, include
3
4  urlpatterns = [
5      path('admin/', admin.site.urls),
6      path('', include('configuration.urls')),
7      path('', include('vms.urls')),
8      path('', include('vmq.urls')),
9      path('', include('gemba.urls')),
10     path('', include('passation.urls')),
11     path('', include('five_s.urls')),
12 ]
```

Figure 4. Fichier urls.py du dossier "maître"

Le fichier urls.py sert à déclarer les sous-fichiers urls.py des applications tierces.

Les fichiers urls.py sont utilisés pour définir les routes de l'application. Il s'agit de fichier Python qui contient une liste de patterns d'URL qui peuvent être mappés à des vues (fonctions ou classes de vues).

Le fichier asgi.py est un fichier utilisé pour définir un point d'entrée pour l'application ASGI (Asynchronous Server Gateway Interface). C'est une interface qui permet de faire communiquer des serveurs asynchrones et des applications Web : il n'a pas été modifié.

Base de données

Les modèles de base de données Django sont définis dans le fichier models.py de chaque application. Ce fichier contient des classes qui héritent de la classe **Model** de Django et qui définissent les champs et les comportements des objets qui seront enregistrés dans la base de données. L'utilisation des commandes Django permet de créer et manipuler les tables de base de données correspondantes et d'effectuer des opérations CRUD sur les données.

Les modèles **BaseModel** et **SoftDeleteModel** sont des modèles de base de données personnalisés dans Django qui permettent de gérer de manière plus efficace les données de l'application.

Le modèle **BaseModel** est un modèle de base de données qui inclut des champs de date de création et de modification par défaut. Cela permet de suivre l'historique des données et de savoir quand elles ont été créées ou modifiées pour la dernière fois.

Le modèle **SoftDeleteModel** est un modèle de base de données qui inclut un champ de suppression logique (*is_deleted*). Au lieu de supprimer physiquement les données de la base de données, le modèle met simplement à jour le champ *is_deleted* pour indiquer que les données sont "supprimées". Cela permet de conserver l'historique des données et de les récupérer s'il y a besoin.

Les modèles **BaseModel** et **SoftDeleteModel** sont utiles lorsque l'on souhaite suivre l'historique des données ou lorsque l'on a besoin de supprimer des données de manière logique plutôt que physique. Pour des raisons de conservations d'historiques, la suppression logique a donc été adoptée.

```
class SoftDeleteModel(models.Model):
    is_deleted = models.BooleanField(default=False)
    deleted_at = models.DateTimeField(blank=True, null=True)
    deleted_by = models.CharField(max_length=50, blank=True, null=True)
    restored_at = models.DateTimeField(blank=True, null=True)
    restored_by = models.CharField(max_length=50, blank=True, null=True)

    objects = SoftDeleteManager()
    all_objects = models.Manager()
    deleted_objects = SoftDeleteManagerDel()

    def delete(self):
        raise Error()

    def soft_deleted(self):
        self.is_deleted = True
        self.deleted_at = timezone.now()
        self.deleted_by = "Default User"
        self.save()

    def restore(self):
        self.is_deleted = False
        self.restored_at = timezone.now()
        self.restored_by = "Default User"
        self.save()

    class Meta:
        abstract = True
```

```
class BaseModel(models.Model):
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)
    created_by = models.CharField(max_length=50, default="Default User")
    updated_by = models.CharField(max_length=50, default="Default User")

    class Meta:
        ordering = ['-updated_at', '-created_at']
        abstract = True

class SoftDeleteManager(models.Manager):
    def get_queryset(self):
        return super().get_queryset().filter(is_deleted=False)

class SoftDeleteManagerDel(models.Manager):
    def get_queryset(self):
        return super().get_queryset().filter(is_deleted=True)
```

Figure 5. Modèles BaseModel et SoftDeleteModel

Ces modèles deux ont été implémenter et leur manager associés ont été mis en place afin qu'ils puissent être utilisés par une grande majorité des autres modèles de l'application.

Une fois les modèles de base spécifiés, les modèles classiques peuvent hérités de ces derniers. Des champs propres au modèle seront alors déclaré et viendront s'ajouter à ceux des modèles de base, leurs managers seront également hérités et approprié par les modèles classiques.

Les modèles division et programmes ont donc été définis comme suit pour exemple :

```
class Division(BaseModel, SoftDeleteModel):
    name = models.CharField(max_length=30)
    location = models.CharField(max_length=30)
    description = models.CharField(max_length=200)

    def __str__(self):
        return self.name

class Program(BaseModel, SoftDeleteModel):
    division = models.ForeignKey(Division, on_delete=models.CASCADE)
    name = models.CharField(max_length=200)
    description = models.CharField(max_length=200)

    def __str__(self):
        return self.name
```

Figure 6. Modèles Division et Programme

Exemple de l'application configuration

L'application configuration permet de mettre en place toute la fondation de l'application en proposant l'implémentation des données de base à travers le frontend comme l'importation de la liste des employés pour que le logiciel puisse avoir accès à leurs informations pour connaître leurs qualifications, leur poste, leur matricule, nom, prénom, s'ils doivent avoir une visite, etc. : une fiche identité qui sera exploitée par le backend.

La configuration propose aussi d'enregistrer les sites de production et les programmes, produits et ateliers correspondant comme l'exige le cahier des charges.

On retrouvera donc la possibilité de créer, modifier, mettre à jour et supprimer (CRUD) chaque modèle pour permettre une gestion optimisée.

```

14 # SPECTO VIEWS - CONFIGURATION :
15
16 > def home(request): ...
17
18
19
20 # CRUD-R for DIVISION
21 def read_division(request):
22     divisions = Division.objects.all()
23     division_count = divisions.count()
24     context = {'divisions':divisions,'division_count':division_count}
25     return render(request,'configuration/division/division.html',context)
26
27
28 > def read_deleted_division(request): ...
29
30
31
32
33
34
35 > def create_division(request): ...
36
37
38
39
40
41
42
43
44
45
46
47 > def update_division(request,division_id): ...
48
49
50
51
52
53
54
55
56
57
58
59
60 > def delete_division(request,division_id): ...
61
62
63
64
65
66
67
68
69 > def restore_division(request,division_id): ...
70
71
72
73
74
75
76
77
78 # CRUD-R for PROGRAM
79
80 > def read_program(request, division_id): ...
81
82
83
84
85
86
87 > def read_deleted_program(request, division_id): ...
88
89
90
91
92
93
94 > def create_program(request, division_id): ...

```

Voici un exemple de CRUD pour les divisions. Chaque fonction gère les actions à effectuer en relation avec le frontend et l'affichage coté utilisateur en question. Ainsi, quand l'utilisateur souhaitera lire les divisions, c'est la fonction "read_division" qui sera appelée.

Comme chaque modèle de fonction, elle prend en argument la requête qui sera envoyé depuis le frontend et retourne simplement la page html à afficher en spécifiant le chemin d'accès du template.

Pour lire les divisions enregistrées, cette fonction fera simplement appel à la base de données via une commande Django (l.22).

Figure 7. CRUD-R pour les modèles division

Le "context" est dictionnaire Python qui stocke les variables souhaitées. Les variables de ce dictionnaire pourront alors être utilisé dans le template. Dans cet exemple, les divisions sont passées au frontend via le "context".

L'identifiant de la division est aussi passé en argument dans les modèles de mise à jour, de suppression et de restauration car la fonction doit savoir de quelle division il s'agit. Cet identifiant sera issu du frontend et dépendra du click de l'utilisateur.

Les arguments supplémentaires sont passés selon les besoins du modèle. Par exemple, les programmes dépendent des divisions donc chaque modèle programme prendra en argument l'identifiant de son modèle mère, en l'occurrence celui de sa division associée.

Un atelier dépend d'un produit, qui dépend d'un programme, qui dépend lui-même d'une division. Les arguments seront donc passés en conséquence.

Le corps du modèle abrite le code à exécuter lorsque ce dernier est appelé, il peut être très simple comme l'exemple "read_division" ou beaucoup plus complexe comme les modèles employés par exemple.

Pour répondre aux attentes, l'importation de la liste des employés depuis un fichier Excel devait être possible car cette liste est fondamentale pour le fonctionnement global de l'application, les employés étant même les utilisateurs de cette dernière. Une correspondance automatique de la division de l'employé, de son atelier de production ainsi que de multiples autres critères a de l'être implémenter en fonction des données déjà présente dans l'application. A chaque importation, des contrôles sont effectués et un grand nombre de tables dans la base de données par exemple sont mises à jour de façon dynamique.

Ce schéma se répète pour l'ensemble des modèles de fonctions qui redirigent chacun vers une vue différente associées à des templates html.

Frontend

Application configuration

Le premier onglet est l'onglet "Employés" et affiche la liste identité de chaque employé. Chaque onglet possède des restrictions d'accès donc seuls les utilisateurs autorisés pourront interagir avec l'application sur ces certains onglets.

The screenshot shows the SPECTO application interface. The top navigation bar includes the SPECTO logo, a search bar, and the user profile 'FRESSON MATHIEU'. The left sidebar lists various modules: Dashboard, Passation, SS, VMS, VMQ, VMQ - KPI, GEMBA WALK, and a CONFIGURATION section with 'Employees' highlighted. The main content area shows the 'Employees' configuration page with a header for 'Mathieu FRESSON, CEO / Co-Founder'. Below this, it states 'EMPLOYEES : 607 registered'. A table with 8 columns is visible, but most of its content is obscured by a large 'Confidentiel' watermark. The visible columns are MATRICULE, NOME, UP, CODE, DÉPARTEMENT, PROGRAM, PRODUCT, and WORKSHOP. Some data is visible in the last few rows of the table.

MATRICULE	NOME	UP	CODE	DÉPARTEMENT	PROGRAM	PRODUCT	WORKSHOP
80004					A320	20VU	Chem 4013/401
80013					Logistique	Magasin	kitting
80036					B.T	NA	NA
80040					NA	NA	NA
80062					NA	NA	NA
80083					Logistique	Magasin	kitting
80093					A320	90VU	T
80103					A320	20VU	TB-4012

Figure 8. Liste identité des employés

Si l'administrateur souhaite mettre à jour la liste des employés, il a la possibilité d'importer un fichier Excel via l'interface suivante :

UPLOAD AN EMPLOYEE DATA FILE

Employee data file*

Choisir un fichier Aucun fichier choisi

CANCEL **UPLOAD**

Une fenêtre s'ouvrira alors et l'utilisateur pourra sélectionner le fichier présent sur son périphérique.

Le chemin d'accès du fichier sera transmis au backend afin qu'une mise à jour de la base de données se fasse ainsi que les différentes correspondances.

L'application configuration propose de nombreuses vues car elle gère aussi les divisions, programmes, produits et ateliers.

Configuration / configuration/division/ SPECTO

Mathieu FRESSON
CEO / Co-Founder

Divisions + Create Division Restore Division

DIVISIONS : 4 registered

NAME	LOCATION	DESCRIPTION	ACTIONS	PROGRAMS
Fouchana	Tunis	Division description	Edit Delete	PROGRAMS
Garden Grove	Etats-Unis	-	Edit Delete	PROGRAMS
Nagoya	Japon	-	Edit Delete	PROGRAMS
Toulouse	France	-	Edit Delete	PROGRAMS

Figure 9. Vue de l'onglet Division

L'utilisateur autorisé peut alors accéder aux programmes du site Latécoère désiré et accèdera alors à la vue suivante.

PROGRAMS : 32 registered for this division

[+ CREATE PROGRAM](#) [↶ RESTORE PROGRAM](#)

DIVISION	NAME	DESCRIPTION	ACTIONS	PRODUCTS
Fouchana	A330	A330	✎ 🗑	PRODUCTS
Fouchana	A320	A320	✎ 🗑	PRODUCTS
Fouchana	Saisie+Archives	Saisie+Archives	✎ 🗑	PRODUCTS
Fouchana	Informatique	Informatique	✎ 🗑	PRODUCTS
Fouchana	Maintenance	Maintenance	✎ 🗑	PRODUCTS
Fouchana	Logistique	Logistique	✎ 🗑	PRODUCTS
Fouchana	LASER	LASER	✎ 🗑	PRODUCTS
Fouchana	Comptabilité	Comptabilité	✎ 🗑	PRODUCTS

Figure 10. Vue de l'onglet Division-Programme

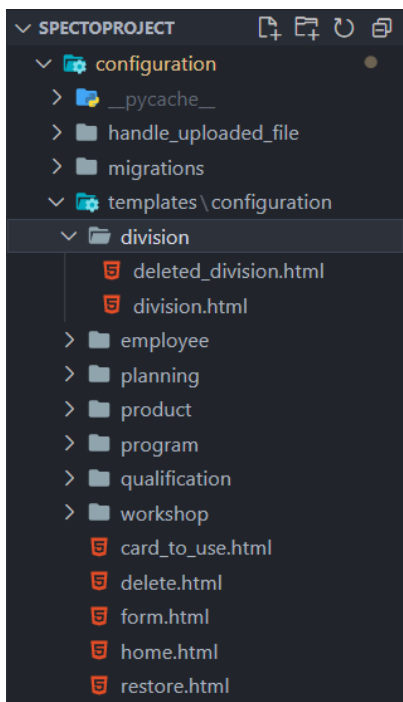


Figure 11. Architecture du dossier templates, app configuration

La procédure se répète pour accéder à la lecture, la modification ou la suppression des produits puis des workshops.

Pour ce qui est de la structure du dossier "templates", il s'agit de fichiers html permettant l'affichage des vues de l'application configuration. Certains fichiers html utilise d'autres fichiers html, appelé composants, qui seront inclus dans les fichiers parents. Cela permet d'optimiser le code sans se répéter.

Les vues issues des fichiers parents, les templates, seront affichées lorsque le backend fera appel à ces derniers.

Chaque application possède un dossier templates qui répertories les fichiers html qui lui sont propre.

Le besoin défini dans le cahier des charges est donc satisfait car l'application web permet maintenant la gestion des employés et celle des divisions et de leurs programmes, produits et ateliers respectifs.

Application VMQ

L'application Visite Managériale Qualité (VMQ) gère l'ensemble des visites managériales en proposant dans un premier temps aux utilisateurs autorisés de pouvoir indiquer la qualification des employés.

Le renseignement de cette qualification permettra au backend de générer les plannings en sachant quels sont les visiteurs (les employés qualifiés) et les visités (les autres).

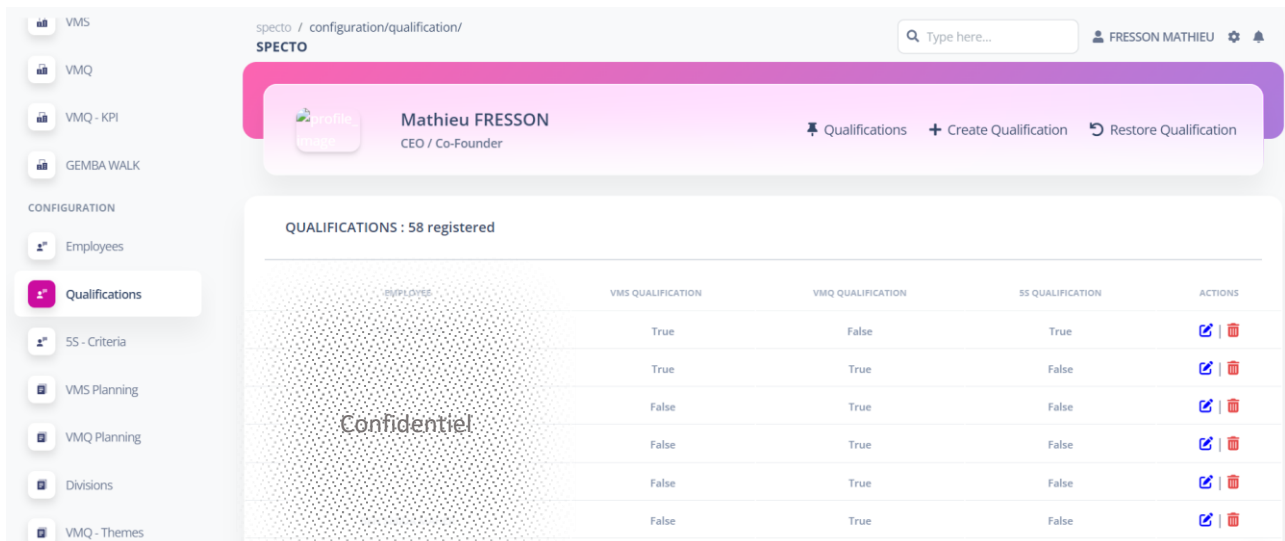


Figure 12. Vue de l'onglet Qualifications

L'ajout d'un employé qualifié se fait en cliquant sur "Create Qualification" et peut être modifié ou supprimé à tout moment.

Une génération du planning VMQ peut alors être faite. Le backend prendra alors en compte les équipes de visites et les dates des dernières visites pour générer le planning VMQ. Les équipes de visites sont composées d'un visiteur qualifié qui visite deux employés. Ces équipes sont indiquées dans la liste identité des employés. Ces nombreux paramètres sont vérifiés et le planning est ajusté automatiquement en conséquence.

Les visites sont mensuelles donc un nouveau planning est généré chaque mois.

DATE	VISITS	GENERATED AT	DETAILS
Dec 2022	46	Nov. 4, 2022, 8:26 a.m.	<button>DETAILS</button>
Jan 2023	46	Nov. 22, 2022, 2:57 p.m.	<button>DETAILS</button>
Nov 2022	44	Nov. 4, 2022, 8:31 a.m.	<button>DETAILS</button>

BACK

Figure 13. Plannings VMQ

Lorsque qu'un planning voit le jour, les visites qualité peuvent alors être effectuées selon les indications. Une fois que le visiteur clôture sa visite, il doit à présent fournir les résultats via un formulaire. Ce formulaire est à remplir depuis l'onglet VMQ puis Create VMQ de l'application. L'ensemble des VMQ seront ensuite listées sur la vue de l'onglet VMQ.

Une fois que les résultats de la visite managériale qualité sont saisies dans l'application, la visite est clôturée pour chaque employé visité dans le planning VMQ du mois correspondant. Le backend ne proposera donc pas ces personnes pour une visite le prochain mois et les proposera de nouveau lorsque l'ensemble des employés auront été visités et que la boucle se réitérera.

DIVISION	DEPARTMENT	MATRICULE	VISITED	PROGRAM	PRODUCT	WORKSHOP	CLOSED	DELETE
	OPEX			A320	106-107-109VU	Equipement-Méca-Inté	True	
	Gestion Industrielle			A320	106-107-109VU	Equipement-Méca-Inté	False	
	Production			A320	106-107-109VU	Equipement-Méca-Inté	True	
Confidentiel	OPEX	Confidentiel		A320	90VU	TB Auto	False	
	Production			LASER	A320/90VU	GAINE	False	
	Qualité			NA	NA	NA	False	
	Gestion Industrielle			A320	90VU	TB ENF	False	
	Gestion Industrielle			LASER	A320/20VU	Impression UV/DEBIT	False	

Figure 14. Exemple de planning VMQ

Lorsque plusieurs visites VMQ ont été réalisés, il y a parfois un certain nombre d'actions à faire à l'issue de ces dernières puisque les possibilités d'amélioration et les choses à corriger ont été identifiées. Un bouton " VMQ Actions" a donc été implémenter afin de pouvoir les consulter facilement et d'y remédier.

KPI

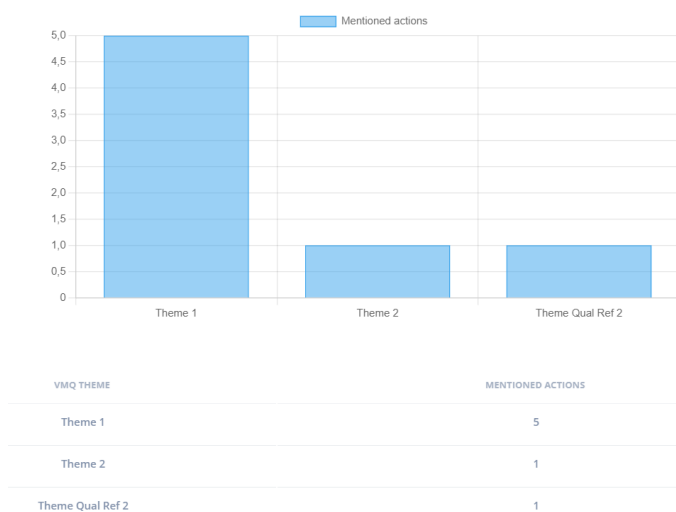


Figure 15. KPI pour les VMQ

Des KPI de ce genre ont été mis en place afin d'identifier rapidement quels thèmes ont le plus d'actions requises et ainsi connaître les secteurs qui font défaut et inversement.

Les KPI permettent une visualisation graphique des données présentes dans toutes l'application, donc d'exploiter sa base et données pour parvenir à des statistiques fiables et pertinents.

Application VMS

L'application pour les visites managériales de sécurité (VMS), le développement se rapproche fortement de celui de l'application VMQ tant sur le backend que sur le frontend. Bien évidemment, les employés qualifiés ne sont pas nécessairement les mêmes et les critères de visites d'une VMQ sont sensiblement différents que ceux d'une VMS mais le principe de génération des plannings, de saisie des formulaires de visites et des qualifications reste le même.

Application 5S

Pour développer l'application sur la technique 5S, la création de checklists pour chaque étape du 5S a été imaginée afin de suivre l'avancement des tâches et de générer des rapports sur les progrès accomplis.

Les critères de cette démarche qualité peuvent être ajoutés, modifiés, supprimés ou restaurés via la CRUD-R disponible sur la vue. Ces critères visent à améliorer l'efficacité et la qualité de l'entreprise en mettant en place un certain nombre de principes de base.

Un onglet de l'application "5S-Criteria" est consacré aux critères de la technique 5S et le frontend se présente comme suit :



Figure 16. Onglet 5S – Critères

Lorsque qu'un contrôle d'application de la technique 5S est effectué, le formulaire à remplir pour y indiquer les résultats est généré automatiquement selon les critères 5S préalablement définis.

L'utilisateur agréé pourra alors compléter ce dernier sans avoir à saisir l'intégralité des critères contrôlés. L'ergonomie a été réfléchie pour permettre un gain de temps quotidien afin d'accroître la productivité de tous : critère primordiale du cahier des charges.

Figure 17. Exemple de formulaires 5S

Les formulaires complétés seront répertoriés dans une liste qui sera affichée sur la vue de l'onglet 5S.

ID	AUDITOR	WORKSHOP	MENTIONED CRITERIAS	ACTION
19	User	Import/export	4	✎ 🗑
18	User	TA-TB	1	✎ 🗑
16	User	Impression UV/DEBIT	1	✎ 🗑
15	User	Impression UV/DEBIT	1	✎ 🗑
14	User	GAINE	1	✎ 🗑

Figure 18. Vue de l'onglet 5S

Application Gemba Walk

L'application Gemba Walk reprends la même architecture de conception que l'application 5S.

Pour développer une application sur la technique de Gemba Walk, des listes déroulantes pour indiquer le statut de chaque critère ont été implémenter afin de suivre la progression et de produire des rapports sur les progrès accomplis.

Un onglet dédié aux critères du Gemba Walk est prévu dans l'application. Lorsqu'un Gemba Walk est effectué, un formulaire est automatiquement généré en fonction des critères préalablement définis. L'utilisateur autorisé peut alors remplir ce formulaire sans avoir à saisir l'intégralité des critères contrôlés. Les formulaires complétés sont répertoriés dans une liste affichée dans l'onglet Gemba Walk de l'application.

Application Passation

L'application passation permet aux employés ayant terminé leurs créneaux de travail d'indiquer l'état de la mission à poursuivre au travers d'un formulaire afin que l'employé relais puisse connaître le travail.

CREATE A NEW PASSATION

Transmitter* Receiver*

ACTION	STATUS	MSN	+/-
Avancement	A faire	e.g : A320	+
Méthode	En cours	e.g : A320	×
Matière	Cloturé	e.g : A320	×

Figure 19. Formulaire de passation

De cette façon, un onglet de navigation est dédié à la passation et présente sous forme de liste l'intégralité des passations de la plus récente à la plus ancienne. Le relais pourra alors facilement identifier la passation du relégué.

Passation

Mathieu FRESSON
CEO / Co-Founder

Passation + Create Passation Restore Passation

PASSATIONS : 4 registered

ID	TRANSMITTER	RECEIVER	ACTION
4	User		Edit Delete
3	User		Edit Delete
1	User		Edit Delete
2	User		Edit Delete

Confidentiel

Figure 20. Onglet Passation

Tests

Les tests, permettent de vérifier que l'application fonctionne correctement et de manière prévisible. Ils permettent de s'assurer que les modifications apportées au code ne cassent pas les fonctionnalités existantes et qu'elles respectent les spécifications définies.

Les tests sont également utiles pour identifier les éventuelles erreurs et bugs dans l'application et pour les corriger avant la mise en production. Ils permettent également de s'assurer que l'application répond correctement aux différentes situations et cas d'utilisation prévus.

En utilisant des tests automatisés, il est possible de vérifier facilement et rapidement que l'application fonctionne correctement après chaque modification apportée, ce qui permet de travailler de manière plus efficace et de livrer des fonctionnalités de qualité. Cependant, des tests classiques ont été effectués pendant ce stage et ces derniers n'ont pas été automatisés car le bon fonctionnement ou non de l'application reste toutefois facile à identifier pour ce type de complexité. Une fois l'application validée et exempte de bugs, la mise en production a été initiée.

Déploiement

Pour déployer une application Django, il est nécessaire de préparer celle-ci en effectuant les tests requis et en s'assurant qu'elle est prête à être mise en ligne. Ensuite, il faut créer un serveur web pour héberger l'application et y installer les dépendances et logiciels nécessaires, tels que Python, Django et les bases de données.

Le code source et les fichiers statiques doivent être transférés sur le serveur, puis les commandes Django nécessaires doivent être exécutées pour mettre à jour les bases de données et collecter les fichiers statiques. Ensuite, le serveur web doit être configuré pour servir l'application Django, puis celle-ci doit être testée pour s'assurer qu'elle fonctionne correctement.

Si tout fonctionne bien, l'application peut être rendue accessible par les utilisateurs en modifiant les paramètres du serveur web et en redirigeant les domaines souhaités vers ce dernier.

Synthèse

Le problème posé était le manque d'efficacité dans la gestion des démarches et procédures de l'entreprise Latécoère, en particulier la rédaction de rapports, que ce soit ceux des visites managériales, ceux des contrôles qualité 5S ou Gemba Walk et la saisie de formulaires sur papier comme ceux de passation, qui nécessitait une analyse manuelle fastidieuse pour exploiter ces données et obtenir les indicateurs clés de performance (KPI) ou encore la génération de plannings qui était plus que chronophages. La solution retenue a donc été la digitalisation de ces procédures afin d'optimiser le temps consacré à ces tâches et de renforcer la sécurité de la mission d'efficacité et la quête de performance de l'entreprise.

La démarche suivie a consisté en une analyse des besoins des utilisateurs et de l'entreprise, ainsi qu'une étude des différentes solutions possibles. Le choix s'est finalement porté sur le développement d'une application web en suivant une méthodologie de conception précise et en effectuant des tests de validation pour garantir la qualité du produit final. Cette solution a été considérée comme la solution la plus adaptée pour répondre aux attentes de toutes les parties et la plus évolutive.

Les résultats obtenus ont été très positifs : l'application de digitalisation mise en place a permis de faciliter les visites managériales, la génération des plannings, le suivi de la démarche qualité tels que le Gemba Walk ou encore le 5S, la gestion des produits, programmes et ateliers selon le site de production, tout cela en un même outil informatique, accessible pour tous et facile à prendre en main. Cela a permis de réduire considérablement le temps nécessaire à la saisie et l'analyse des données, ce qui a entraîné une augmentation de l'efficacité des employés et donc de l'entreprise Latécoère.

En ce qui concerne les perspectives, il y a de fortes chances que l'application de digitalisation soit adoptée à grande échelle par l'entreprise, ce qui pourrait entraîner une augmentation de la productivité et de la qualité de l'entreprise dans les années à venir. De plus, l'application peut être facilement mise à jour et évoluée en fonction des besoins de l'entreprise, ce qui offre de nombreuses possibilités de développement futur.

Conclusion

Conclusion technique

Pour répondre aux besoins de l'entreprise Latécoère en matière de digitalisation de ses processus de visites managériales, de génération de plannings et de suivi de la démarche qualité, une solution a été apportée, désignée comme celle étant la plus adaptée et optimisée pour ces attentes exigeantes : le développement d'une application web full stack.

La méthodologie de conception adoptée pour ce projet a été le modèle en V, avec une phase de recueil des besoins et une phase de conception en amont du développement proprement dit. Les technologies sélectionnées pour le développement ont été l'utilisation de HTML, JavaScript et CSS pour le front-end et le framework Django en Python pour le back-end, avec une base de données PostgreSQL pour stocker les données.

Le développement de backend en Python a été sélectionné pour sa souplesse et sa vitesse de création. Le framework Django simplifie la mise en place de backend en Python en proposant des outils pratiques pour gérer les routes, les bases de données et les utilisateurs et propose des outils très efficaces pour la liaison avec le frontend.

Les étapes de développement de l'application ont comporté la mise en place de l'architecture de l'application, l'implémentation des fonctionnalités et l'intégration de l'interface utilisateur. Une stratégie de tests de validation a été mise en place pour s'assurer de la qualité et de la fiabilité de l'application avant sa mise en production.

Le projet d'étude a été mené à terme avec succès et les délais (ceux de la durée du stage) ont été respectés, l'application a été déployée et mise en production pour être utilisée par les employés de Latécoère. Les résultats obtenus ont montré une augmentation de l'efficacité des processus de l'entreprise grâce à la digitalisation proposée, ainsi qu'une amélioration de la qualité des produits grâce au suivi en temps réel de la démarche qualité.

Conclusion personnelle

Le stage que j'ai réalisé chez Latécoère a été une expérience très enrichissante, tant sur le plan technique que sur le plan professionnel. J'ai eu l'opportunité de mettre en pratique mes connaissances en développement web et de découvrir de nouvelles technologies et méthodologies de travail.

Le projet de digitalisation que j'ai mené a été une réelle opportunité pour moi de mettre mes compétences en application et de contribuer concrètement à l'amélioration des processus de l'entreprise. J'ai pu développer une application web complète et fonctionnelle, répondant aux besoins exprimés par les managers et les employés de Latécoère, et qui a été très appréciée lors de sa mise en production.

Ce stage m'a également permis de découvrir le fonctionnement d'une entreprise de l'aéronautique et de travailler en étroite collaboration avec le responsable IT du site de Tunis, mon tuteur de stage. J'ai su travailler en autonomie et gérer mon projet de manière efficace, grâce à l'accompagnement de mon tuteur et de certains collègues. Des connaissances approfondies en informatique procure des opportunités de projets ou business dans de nombreux domaines, les limites n'étant que celles que l'on se fixe.

Ce stage aura été une expérience très positive et me conforte à l'idée de poursuivre mon parcours professionnel dans ce domaine que je trouve passionnant.

Bibliographie

- [1] Latécoère, "Présentation de l'entreprise", (2019), [En ligne]. Disponible sur : <https://www.latecoere.com/fr/presentation/presentation-de-lentreprise>.
- [2] Normes aéronautiques, "Les normes aéronautiques et spatiales", (2020), [En ligne]. Disponible sur : <https://www.aeronautique-defense.gouv.fr/les-normes-aeronautiques-et-spatiales>.
- [3] C. Naud, "Le Gemba Walk, une méthode de management de la qualité", (2019), [En ligne]. Disponible sur : <https://www.lesechos.fr/2019/07/09/le-gemba-walk-une-methode-de-management-de-la-qualite-1177963>.
- [4] B. Depaire, "Le 5S, une méthode de gestion visuelle pour améliorer la performance de l'entreprise", (2018), [En ligne]. Disponible sur : <https://www.usinenouvelle.com/article/le-5s-une-methode-de-gestion-visuelle-pour-ameliorer-la-performance-de-l-entreprise.html>.
- [5] A. Lecointre, "La gestion de production : définition et enjeux", (2020), [En ligne]. Disponible sur : <https://www.gestiondeproduction.com/definition/>.
- [6] P. Joly, "Définition de l'ergonomie", (2020), [En ligne]. Disponible sur : <https://www.linternaute.com/dictionnaire/fr/definition/ergonomie/>.

Annexes

Aucune annexe n'a été jugée suffisamment pertinente pour être inclus dans ce rapport.