

PROJET D'ARCHITECTURE

Analyse de faisabilité

Rep'Aero



24 mars 2021

SOMMAIRE

<u>Section</u>	<u>Page</u>
1. INTRODUCTION	3
1. ARCHITECTURE	4
1.1. Architecture actuelle.....	5
1.2. Architecture cible	7
2. GESTION DES PARTIES PRENANTES.....	9
2.1. Identification des parties prenantes	9
2.2. Analyse des parties prenantes	10
2.3. Points forts et faiblesses	10
2.4. Comportement s'il y a des problèmes	11
3. CONTRAINTES.....	12
3.1. Identification des contraintes	12
3.2. Comment traiter les contraintes.....	13
4. RISQUES.....	14
4.1. Registre de risques	14
4.2. Matrice de risques	15
5. STRATEGIE POUR LIMITER LES IMPACTES.....	16
6. CONCLUSION	17
7. ANNEXE	18
7.1. Architecture existante.....	18
7.2. Architecture cible.....	19
7.3. Matrice de risques	20

1. INTRODUCTION

Rep'Aero



Rep' Aero travaille avec ses clients pour assurer la maintenance corrective et préventive des pièces d'avion sous sa responsabilité (y compris les pièces motorisées, structurelles et avioniques).

L'entreprise joue un rôle clé dans le maintien de la navigabilité des avions, en garantissant le suivi sécurisé des données constructeurs, ainsi que l'application des réglementations françaises et européennes.

Cette analyse de faisabilité décrit les résultats sur la migration de l'architecture, les besoins du projet, le coût financier, les scénarios possibles et le scénario le plus adapté.

Dans les pages suivantes, nous trouverons les détails tout au long de cet analyse.

1. ARCHITECTURE

L'architecture est la structure générale inhérente à un système informatique, l'organisation des différents éléments du système (logiciels et/ou matériels et/ou humains et/ou informations) et des relations entre les éléments. Cette structure fait suite à un ensemble de décisions stratégiques prises durant la conception de tout ou partie du système informatique, par l'exercice d'une discipline technique et industrielle du secteur de l'informatique appelée elle aussi l'architecture.

La structure d'un système informatique est représentée sous forme de graphiques tels que des organigrammes, des diagrammes de workflow ou des diagrammes entité-relation. Le diagramme peut concerner un logiciel, une pièce de matériel, un réseau informatique, un groupe de machines, un sous-système, voire l'ensemble des dispositifs informatique d'une entreprise ou d'une institution.

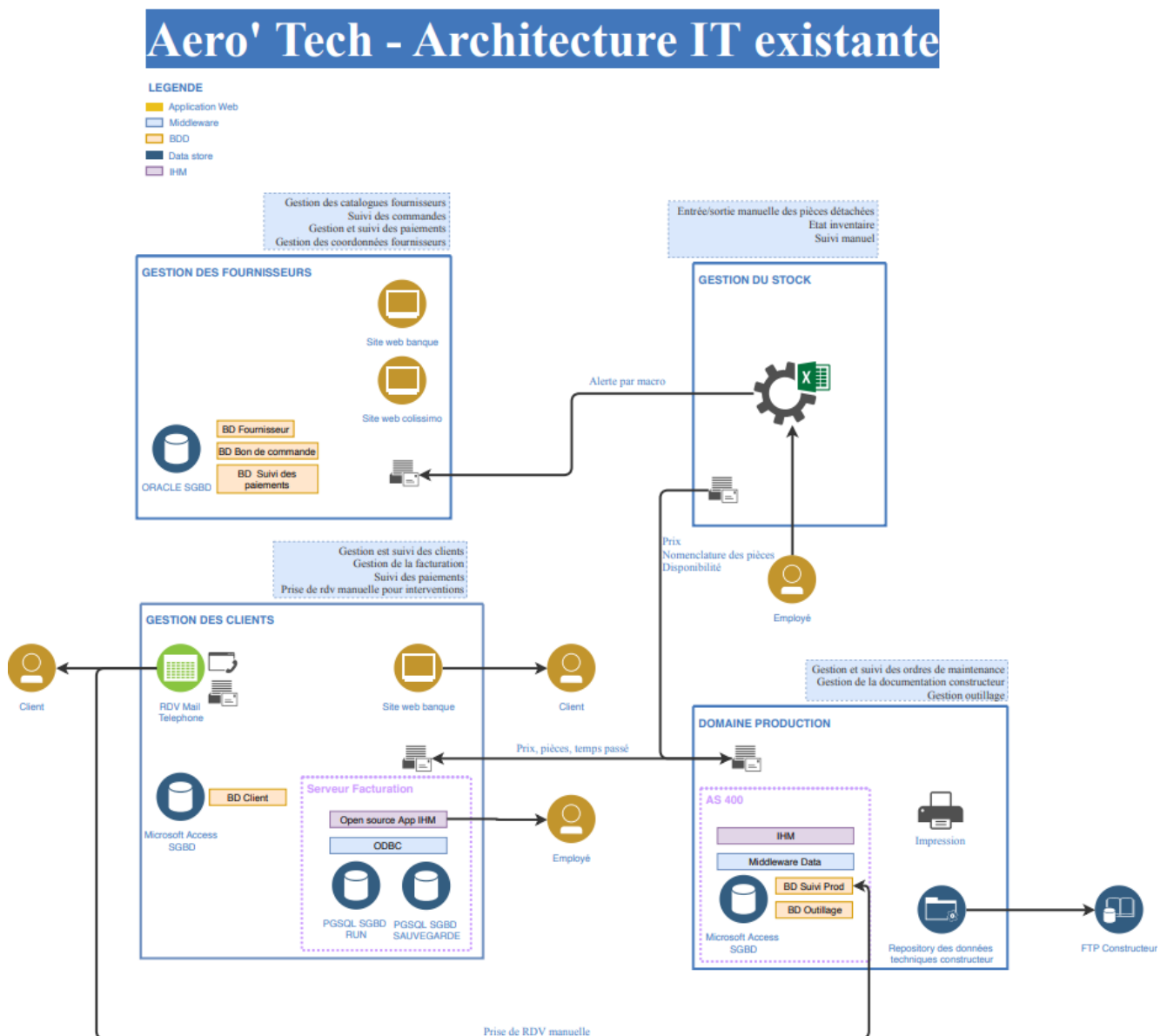
Un diagramme d'architecture est une perspective qui dépend du point de vue adopté par son auteur, en fonction des éléments qu'il recherche à mettre en évidence. Le diagramme omet volontairement certains détails pour rendre la perspective plus visible. Il peut y avoir plusieurs diagrammes d'architecture pour un même système, tels que : architecture des informations, architecture métier, architecture applicative et architecture technique.

- **L'architecture métier** est un point de vue tourné sur les politiques, les stratégies et les procédures opérationnelles propres à une organisation et les différents éléments du système informatique en rapport avec eux.
- **L'architecture des informations** est un point de vue tourné sur l'organisation, le classement et la présentation des informations propres à une organisation en accord avec la manière dont les gens vont interpréter, retrouver ou modifier ces informations.
- **L'architecture logicielle** est une vue tournée sur l'organisation interne et le découpage en couches et modules du ou des logiciels du système informatique. Les responsabilités de chaque module et la nature et la structure des relations entre modules.
- **L'architecture technique** est une vue tournée vers les différents éléments matériels et l'infrastructure dans laquelle le système informatique s'inscrit, les liaisons physiques et logiques entre ces éléments et les informations qui y circulent.
- **L'architecture matérielle** est une vue tournée sur le choix et l'organisation des différents composants électroniques d'un appareil informatique.

1.1. Architecture actuelle

En rassemblant les informations présentées par l'entreprise, l'architecture actuelle présente plusieurs problèmes de performance, de qualité de service avec des conséquences importantes reflétées dans les résultats : la perte de clients.

Dans l'image suivante nous trouverons le schéma de l'architecture actuelle :



(Voir Annexe 7.1)

En rassemblant les informations présentées par l'entreprise, l'architecture actuelle présente plusieurs problèmes de performance, de qualité de service avec des conséquences importantes reflétées dans les résultats : la perte de clients.

Ensuite, nous détaillerons les problèmes déjà mentionnés ci-dessus et leurs conséquences dans le but de les expliquer.

Performance :

- L'utilisation de technologies obsolètes.
- Hétérogénéité des systèmes de persistance.
- Utilisation de technologies coûteuses et non personnalisables.
- Mise à jour lente et compliquée.

Qualité du service :

- Redondance de certaines opérations d'entrée.
- Mauvaise intégration du système
- Prolifération des doublons dans les bases de données
- Manque d'interopérabilité avec les systèmes clients
- Flux d'informations non standardisés
- Manque de contrôle dans le téléchargement des informations.

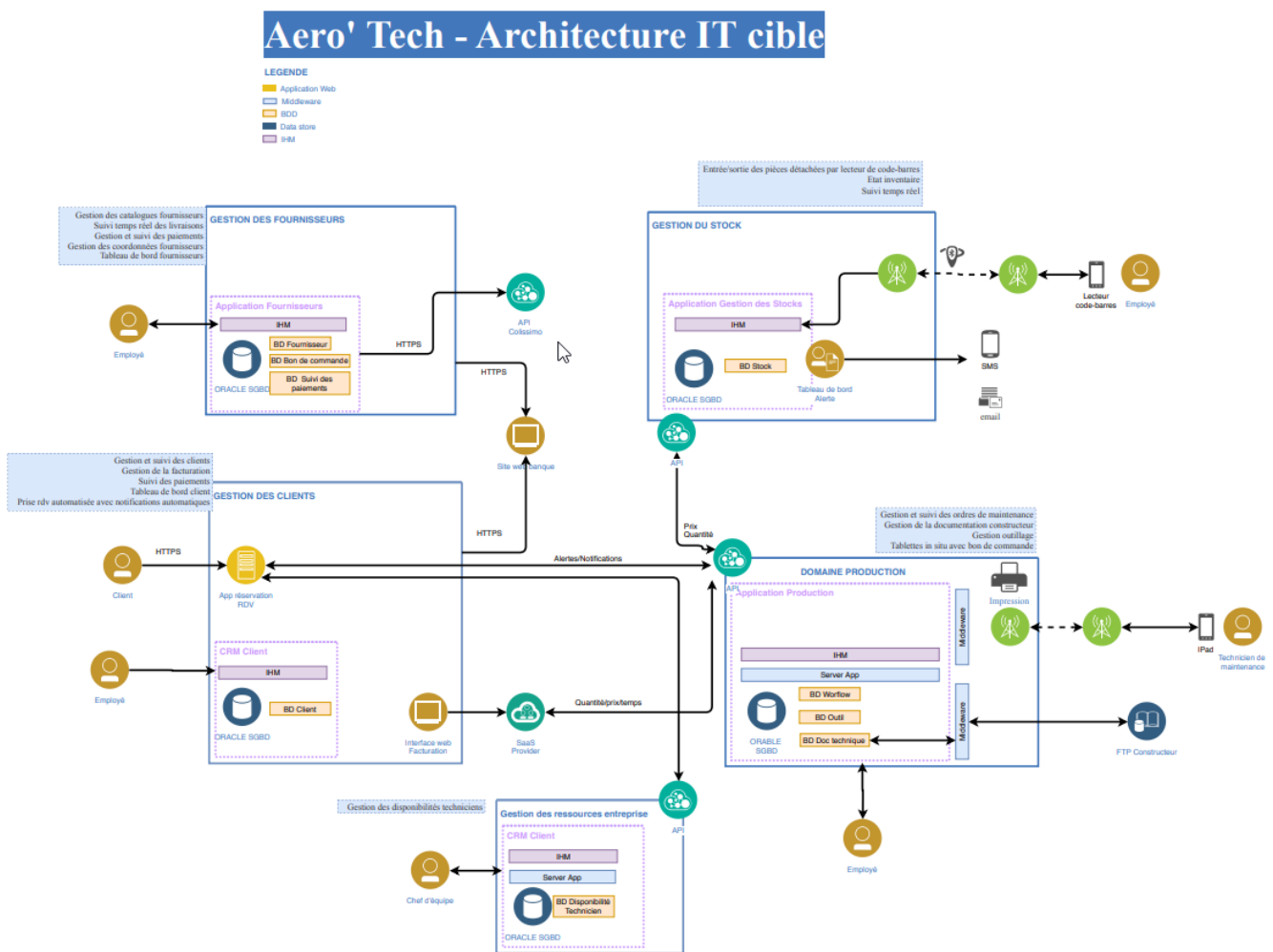
Cela a les **conséquences** suivantes :

- Manque de capacité de réponse
- Manque de motivation des employés
- Émergence de processus de gestion parallèles
- Difficulté de suivi financier et de suivi des stocks

1.2. Architecture cible

L'architecture cible comprend de nouveaux outils qui faciliteront le travail des employés et le processus pour les clients en réduisant le nombre d'étapes du système, ainsi qu'en automatisant et en mettant à jour les données en temps réel.

Dans ce qui suit, les grandes lignes de l'architecture cible seront présentées, puis les aspects positifs de ces changements seront détaillés en profondeur.



(Voir Annexe 7.2)

Pour continuer, il est nécessaire de passer en revue chacun des domaines :

- **Gestion du stock** : Le changement radical consistant à passer d'un enregistrement manuel de chacun des éléments qui composent le stock à un enregistrement automatique par capture de codes-barres rationalise le processus puisque la base de données est maintenue à jour et prête à être lue en permanence. L'activation des alertes de suivi des stocks libère l'employé des actions manuelles dans le système. Tout cela à partir de l'application dédiée à la gestion des stocks.
- **Gestion des fournisseurs** : Pour commencer, on peut voir que le fait de développer une application pour ce domaine puisqu'il comprend la base de données des fournisseurs, la base de données des commandes et les documents liés au suivi des paiements. Un autre changement notable s'observe dans l'automatisation des paiements et des encaissements grâce à l'intégration du site web de la banque via une connexion HTTPS avec laquelle les clients peuvent payer directement en ligne. De même, la mise en œuvre de l'API Colissimo rendra les envois plus simples et plus rapides, puisque le système effectuera les requêtes automatiquement.
- **Gestion des clients** : L'application est développée pour cette partie de l'entreprise qui permettra au client d'avoir un portail pour fixer des rendez-vous, demander de nouvelles pièces. Le fait d'automatiser les réservations accélère le processus. Pour la partie interne, l'employé a accès à la base de données contenant les informations sur les clients et à l'interface web pour la facturation, de sorte que le nombre d'actions requises pour le processus de travail habituel est réduit.
- **Gestion des ressources de l'entreprise** : il s'agit d'une application simple mais utile dans laquelle le chef d'équipe peut voir la disponibilité de chacun des techniciens qui confirment son équipe de travail. Il est connecté à l'application de gestion des clients afin que la base de données soit toujours mise à jour et que les erreurs dans le plan de travail soient évitées.
- **Zone de production** : l'application est mise en œuvre pour cette zone qui contient la base de données des plans de travail, la base de données de contrôle des outils, ainsi que la documentation technique. Les techniciens de maintenance pourront imprimer les commandes en fonction de la situation, ainsi que la demande de nouvelles pièces automatiquement au fournisseur.

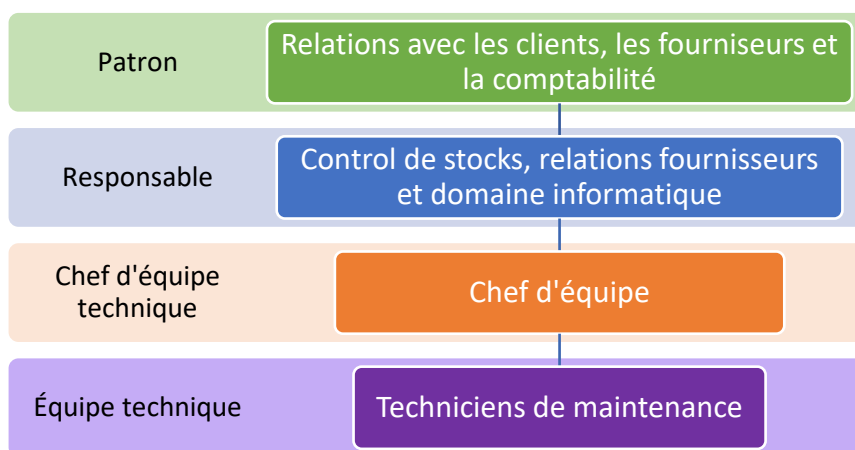
Pour conclure l'analyse de l'architecture cible, nous pouvons conclure avec les aspects positifs qui se refléteront pendant le processus de travail, le système sera connecté, mis à jour et disponible à tout moment, les clients auront un processus plus facile ainsi que les employés. Cela se traduit par une efficacité, une rapidité et une disponibilité de tous les instants.

2. GESTION DES PARTIES PRENANTES

Avant d'entrer dans le détail et analyse des parties prenantes, nous allons définir ce concept. Les parties prenantes d'un projet sont les personnes, les groupes ou les organisations susceptibles d'affecter le projet ou d'être affectés par celui-ci. C'est-à-dire qu'il s'agit de tous les acteurs qui ont de l'influence sur le cours du projet. Cela comprend donc ceux qui peuvent favoriser la réussite comme ceux qui peuvent freiner l'avancement du projet.

2.1. Identification des parties prenantes

La liste initiale des participants est définie en tenant compte de l'équipe de projet initial, cependant, elle peut changer en fonctions des besoins du projet :



En analysant les premiers membres de l'équipe, le tableau suivant a été établi pour les décrire et les actions à mener en fonction du degré d'influence sur le projet.

FONCTION	INFLUENCE	ACTION
Patron	5	MANTENIR SATISFAIT
Responsable	4	GÉRER AVEC ATTENTION
Chef d'équipe technique	3	TENIR INFORMÉ
Développeurs	2	SURVEILLER
Équipe technique	2	SURVEILLER

2.2. Analyse des parties prenantes

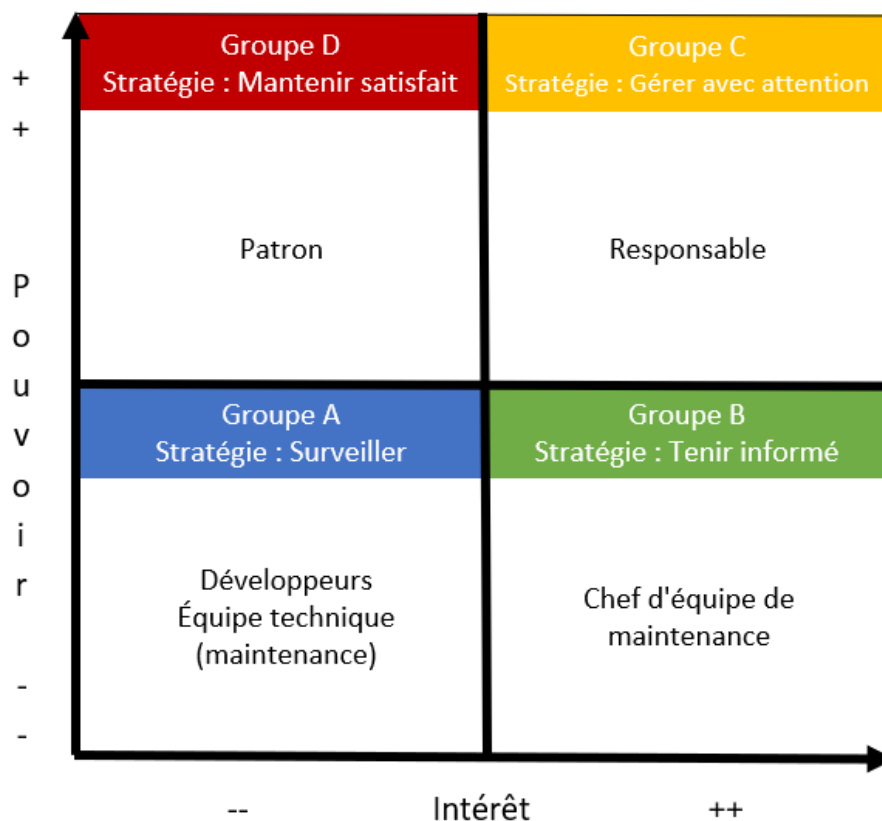
En plus de l'équipe initiale, une équipe de développeurs spécialisés dans les différentes branches sur lesquelles les changements seront appliqués sera nécessaire, c'est-à-dire une équipe externe qui sera en charge du processus de développement et de migration vers l'architecture cible. Cette équipe sera définie comme les parties prenantes du projet.

Les profils nécessaires sont :

- Un développeur Front-end.
- Un développeur Back-end.
- Un développeur spécialisé dans les bases de données.
- Un technicien spécialisé en réseaux de communication.
- Un testeur.

2.3. Points forts et faiblesses

En continuant avec le tableau présenté ci-dessus, la matrice suivante a été réalisée, qui met en relation le niveau d'influence avec celui du pouvoir, car il est très important de voir dans quelle mesure chaque membre peut affecter le projet de manière positive ou négative.



L'influence se réfère au pouvoir que les parties prenantes ont sur le projet. Elle peut être exercée en contrôlant directement le processus d'élaboration de la décision et en facilitant ou gênant la mise en œuvre du projet. Ce contrôle peut résulter du statut ou du pouvoir que détient une partie prenante ou provenir de relations informelles avec des leaders.

2.4. Comportement s'il y a des problèmes

Il convient de préciser que les acteurs internes du projet, c'est-à-dire l'employeur et les employés, ne seront pas en mesure de réagir à un problème technique car ils n'ont pas la capacité de le résoudre. Si des problèmes techniques surviennent, ce sera à l'équipe qui sera chargée de cette partie.

C'est-à-dire les problèmes liés à la migration et à la création des bases de données, avec les codes du nouveau système, les IHM, les connexions que le système aura et la manière dont les nouveaux composants du système communiqueront. Ils doivent être clairs sur les actions nécessaires pour résoudre ces conflits, mais ils auront toujours certaines limites en termes de prise de décision puisqu'ils doivent transmettre ces préoccupations au patron, car c'est lui qui a le pouvoir de décision.

L'une des stratégies et des avantages de travailler avec un contrôle de version tel que Git, est que si nous trouvons des problèmes au moment des tests unitaires ou fonctionnels, nous avons la possibilité de revenir à une version antérieure afin d'éviter les erreurs, les échecs et autres situations négatives par rapport à leur travail.

Cependant, il est important, comme mentionné ci-dessus, de communiquer toute situation ou préoccupation au responsable ou à l'employeur de l'entreprise afin de recevoir les indications correspondantes.

Quant au manager, il est responsable des actions liées à la communication et aux préoccupations des clients, puisque c'est lui qui est en relation avec eux. Tout problème annoncé par l'équipe de développeurs sera répercuté dans l'équipe interne au moment de le communiquer aux clients.

L'équipe de maintenance, doit communiquer ses préoccupations à son supérieur immédiat, afin que tous les doutes concernant le système et le nouveau fonctionnement lui soient expliqués, ainsi que signaler les défaillances des applications de quelque nature que ce soit.

3. CONTRAINTES

Une contrainte peut se définir comme un facteur qui limite la performance globale d'un projet ou d'un système. Cette contrainte va par exemple limiter la capacité d'une équipe à atteindre l'objectif qui lui a été fixé.

Elle peut agir comme un goulot d'étranglement limitant les capacités de production. Afin de mieux repérer les contraintes, des analyses basées sur des observations et des mesures sont nécessaires.

3.1. Identification des contraintes

- **Les coûts**

Lors de la phase de planification d'un projet, il y a toujours un risque d'avoir oublié un concept qui pourrait mettre en péril le budget fixé pour atteindre les objectifs. C'est pourquoi, avant de commencer un projet, il est conseillé de prendre en compte de manière exhaustive toute activité, salaire des employés, coût du matériel, et tout autre concept qui génère une charge économique pour le projet. Par exemple:

- Temps consacré aux réunions, aux rapports, etc.
- Temps consacré au développement informatique du projet.
- Temps passé à tester le développement du logiciel.

Il est important d'être aussi précis que possible car il sera consommé au fur et à mesure de l'avancement du projet. Et ce n'est que lorsque tous les participants ont examiné et validé le budget que le projet peut démarrer, sans la validation de tous les participants n'est pas possible.

- **Les délais**

Un planning précis doit être établi entre la maîtrise d'œuvre et la maîtrise d'ouvrage. Ce planning doit donner les dates jalon principales, c'est à dire celles qui correspondent à des étapes précises dans le projet.

La validation de ce planning est importante, parce qu'il sera utilisé par toutes les parties pour juger de l'avancement du projet. Ce planning doit être établi en tenant compte de tous les paramètres pouvant impacter le projet : congés, ressources disponibles, délais incompressibles de certaines actions, etc.

La maîtrise d'œuvre risque de ne pas pouvoir respecter ses engagements en matière de délais si elle se voit contrainte d'adapter le planning non pas en fonction des véritables contraintes, mais en fonction de la demande initiale de la maîtrise d'ouvrage qui souhaite un délai très court, au risque que ce délai soit impossible à tenir.

- **La qualité**

Un développement informatique répond à des règles de l'art précis qui obligent la maîtrise d'œuvre à livrer à sa maîtrise d'ouvrage un outil informatique qui fonctionne sans erreurs, et surtout, qui respecte le cahier des charges fonctionnelles validées avec la maîtrise d'ouvrage.

La maîtrise d'œuvre risque de ne pas pouvoir tenir ses contraintes de qualité si la maîtrise d'ouvrage modifie en cours de projets ses spécifications fonctionnelles, en ajoutant ici et là des fonctionnalités impactantes, non prévues initialement.

3.2. Comment traiter les contraintes

Le principe consistera donc plutôt à étudier et analyser la contrainte afin d'en déterminer les effets exacts, et surtout les causes qui pourront ensuite être traitées.

Toutes les contraintes ne pourront pas nécessairement être éliminées. Néanmoins, le but est de :

- Minimiser leur impact.
- Améliorer ainsi les performances globales du système ou du projet.
- Récupérer éventuellement des ressources qui pourront être utilisées à d'autres tâches.

L'analogie de la chaîne est tout à fait adaptée à notre cas. La solidité d'une chaîne est égale à celle de son maillon le plus faible. On peut donc voir les différentes tâches répertoriées dans un projet comme des maillons d'une chaîne. Si une contrainte s'applique à l'une de ces tâches (contrainte technique, humaine...), la performance globale du projet est limitée par cette contrainte.

L'élimination ou la diminution de l'impact de cette contrainte permettra de renforcer automatiquement l'ensemble de la chaîne.

L'identification des contraintes et leur élimination est un processus continu durant toute la durée du projet. Il s'agit de régulièrement réévaluer les contraintes ainsi que leurs impacts. L'identification, l'exploitation et la suppression d'une contrainte amène à l'amélioration des performances, puis à la recherche de nouvelles contraintes. Nous sommes donc clairement ici dans un processus d'amélioration continue.

4. RISQUES

L'analyse des risques est une discipline qui a pour but d'étudier toutes les remarques et alertes critiques soulevées avant et pendant le projet dans le but de mener celui-ci à son terme. Découvrons ensemble de quelle manière identifier et caractériser le risque afin de choisir un modèle de projet.

De manière théorique, il n'est pas véritablement possible de définir avec précision les objets qui peuvent se révéler des risques pour un projet. En effet, les raisons de l'échec sont intrinsèquement liées aux circonstances et aux caractéristiques spécifiques du projet en question.

4.1. Registre de risques

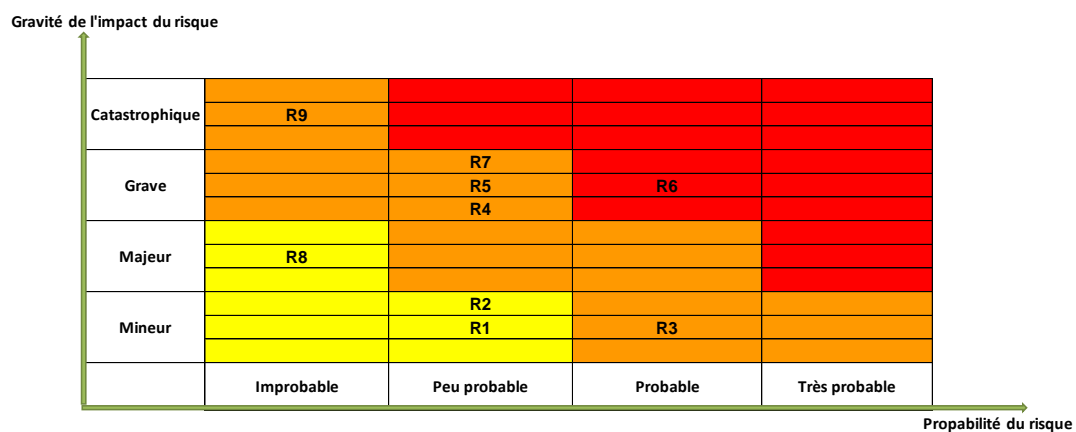
Tout type de risque qui pourrait affecter ou impacter le projet de manière négative est pris en compte pour l'enregistrement.

ID	Description	Gravité	Probabilité	Solution
R1	Frictions au sein de l'équipe projet	Mineur	Peu probable	Définir clairement et précisément les rôles et les tâches de chacun dans le projet
R2	Spécifications ambiguës	Mineur	Peu probable	Faire une explication exhaustive à l'équipe
R3	Un des membres de l'équipe perd sa motivation	Mineur	Probable	Mettre en place des meetings hebdomadaires pour partager les avancées de tous
R4	Peu d'expertise des programmeurs	Grave	Peu probable	Préparer des formations sur les technologies que nous utiliserons
R5	Budget et délais insuffisants	Grave	Peu probable	Laisser une marge dans le planning et budget

R6	Les machines nécessaires au projet se révèlent indisponibles au moment nécessaire	Grave	Probable	Planifier la réservation des machines nécessaires auprès de l'atelier/du service concerné à l'avance
R7	Indisponibilité des fournisseurs informatiques	Grave	Peu probable	Avoir des méthodes prêtes pour consommer les APIs des fournisseurs
R8	Perte de données ou données dupliqués	Grave	Probable	Mettre en place un système de filtrage et aussi garder la base de données d'origine si jamais il faut corriger les doublons
R8	Fuite d'information	Majeur	Improbable	Donner des indications et outils nécessaires à l'acteur que s'occupera
R9	Serveur en panne.	Catastrophique	Improbable	Avoir un deuxième serveur en cas d'urgence

4.2. Matrice de risques

Après avoir traité les informations contenues dans le registre des risques, une matrice des risques a été élaborée, qui est présentée ci-dessous :



(Voir Annexe 7.3)

5. STRATEGIE POUR LIMITER LES IMPACTES

Chaque projet comporte des risques et des limites, de tels concepts peuvent entraîner des complications et des impacts qui peuvent donner des résultats négatifs à différents niveaux. Elles peuvent aller de légères à catastrophiques, il est donc important d'établir une stratégie contenant un plan d'action au cas où cela serait nécessaire.

L'une des façons de se préparer à toute situation problématique est d'avoir une bonne communication entre les membres de l'équipe, afin d'éviter les malentendus sur les questions liées au projet.

Il est important que les acteurs internes de l'entreprise expriment toutes leurs préoccupations et leurs besoins avant le début du projet, car une fois celui-ci lancé, des modifications peuvent être apportées, mais pas de manière aussi radicale car elles mettraient en péril la qualité et le délai de livraison.

Il existe différents outils et différentes manières de procéder à un changement architectural aussi important que celui qui sera effectué à cette occasion. Par exemple, pour obtenir un travail fluide et agile, nous avons comme option le développement et l'intégration continus pendant la phase technique, de cette façon l'équipe technique réduira le nombre de tâches pour se concentrer sur ce qui est le plus important. Le code est maintenu à jour et la livraison peut être automatisée. Même la partie test est incluse dans ce système de travail, juste avant la livraison, les acteurs qui seront impliqués dans cette partie auront des outils pour travailler et éviter tout type de retard et de problème.

L'adoption d'un bon processus de mise en œuvre nous permettra de modifier l'architecture sans mettre le système en danger, c'est-à-dire que les services resteront en ligne et que les clients n'auront aucun problème pour saisir leurs demandes.

Une chose très importante qu'il faut souligner est l'entrée et la mise en œuvre des nouvelles bases de données, puisque nous en avons déjà une, qui n'est pas faite de manière optimale et recommandée, il faudra donc faire un filtrage des données en double.

6. CONCLUSION

Au vu de toutes les informations recueillies et présentées dans ce document, on peut conclure que la mise en œuvre d'une nouvelle architecture est tout à fait possible et réalisable. Il convient de prêter attention au budget, car une équipe externe supplémentaire sera nécessaire pour la partie technique et tous les changements requis pour atteindre les objectifs de ce projet.

Ce projet aura un résultat positif puisque les problèmes internes qui ont entraîné la perte de clients importants pourraient être résolus, de même que la lenteur, l'inefficacité et l'inexactitude avec lesquelles les données sont saisies seront améliorées avec la mise en œuvre de cette nouvelle architecture puisqu'elle est automatisée.

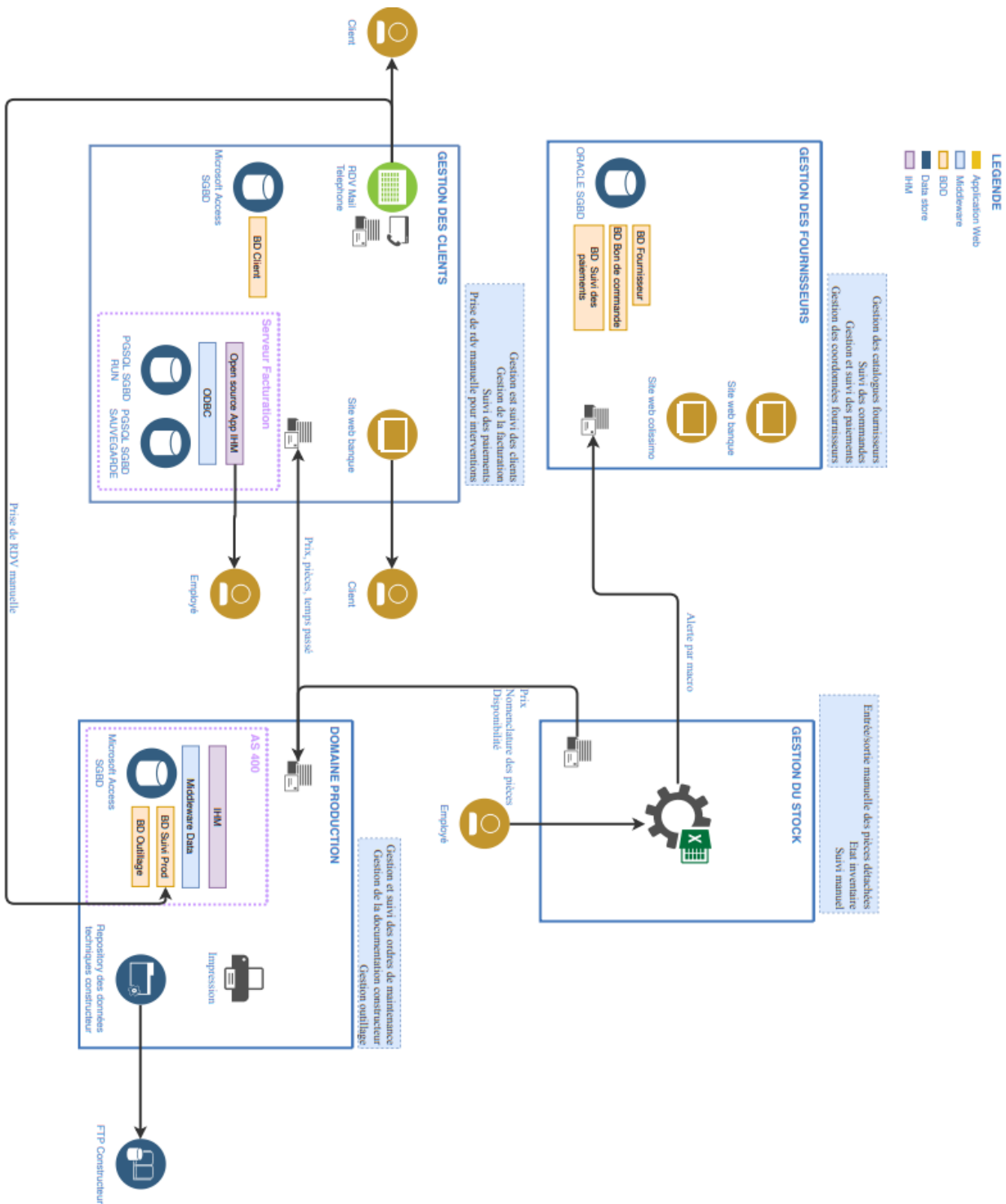
Les clients disposeront d'un portail d'application pour demander des services, des pièces détachées et fixer des dates pour les rendez-vous d'entretien, ce qui améliorera l'expérience des utilisateurs, ce qui ne peut être que positif pour les deux parties.

De même, les délais de développement et de mise en œuvre doivent être pris en compte pour éviter les retards et l'augmentation des coûts lorsque ces concepts sont liés.

De plus, une formation est nécessaire pour les employés afin de leur transmettre le nouveau mode de fonctionnement du système cible, car cette formation prendra du temps.

En résumé, compte tenu de tous ces points, on peut conclure qu'il est possible d'aller de l'avant avec un tel plan.

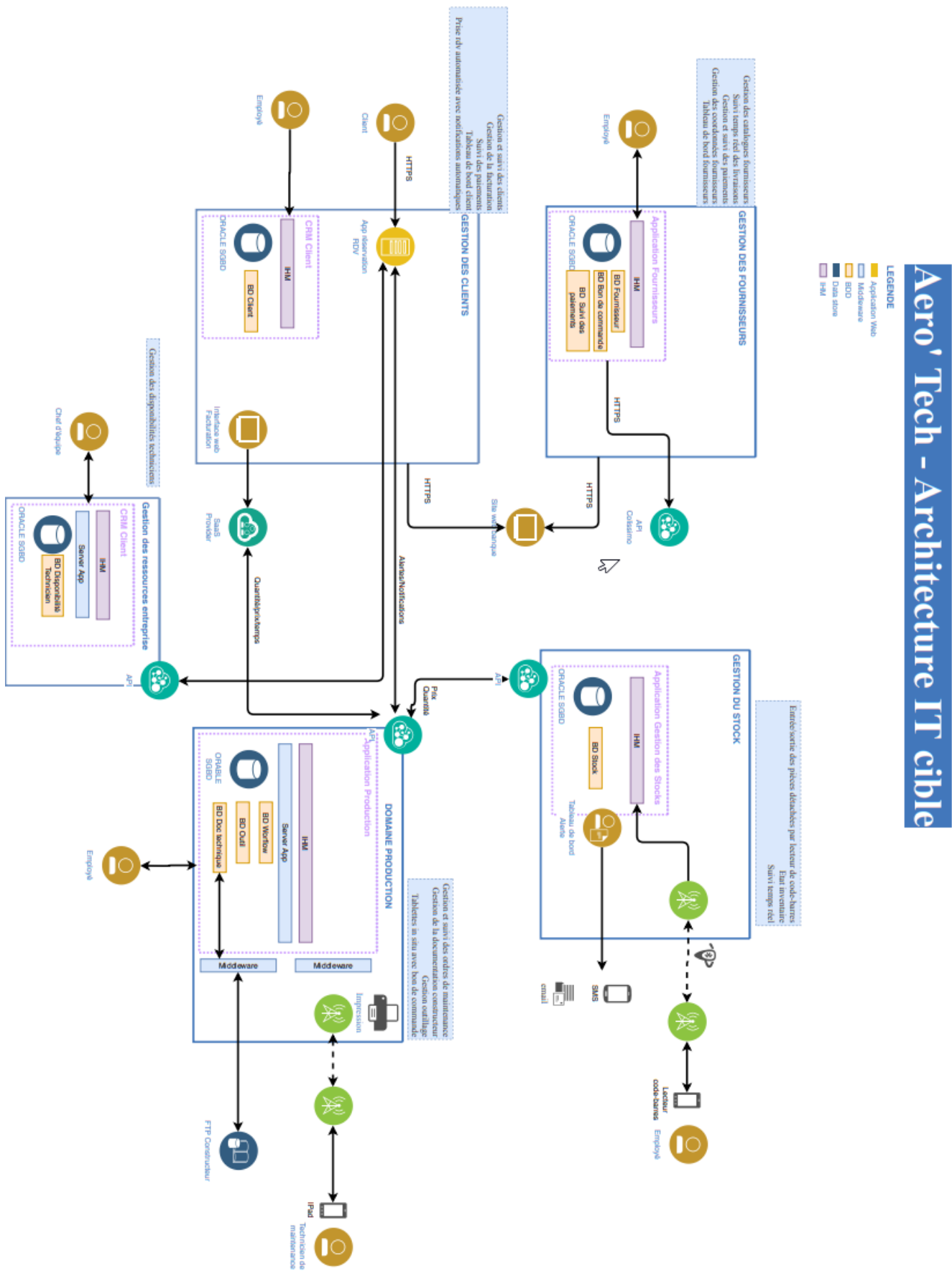
Aero' Tech - Architecture IT existante



7. ANNEXE

7.1. Architecture existante.

7.2. Architecture cible.



7.3. Matrice de risques

