

REPUBLIQUE DU CAMEROUN

Paix-Travail-Patrie

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR

Université de Yaoundé

Institut Universitaire Saint Jean

Saint Jean Ingénieur



UNIVERSITÉ SAINT JEAN
SAINT JEAN INGÉNIEUR

REPUBLIC OF CAMEROON

Peace-Work-Fatherland

MINISTRY OF THE HIGHER EDUCATION

University of Yaoundé

Saint Jean University Institute

Saint Jean Ingénieur



Rapport de Spécification Technique et Fonctionnelle du Système de Monitoring APS

**THEME: MISE EN PLACE D'UN SYSTÈME DE
MONITORING APPLICATIVE D'ALFRESCO PROCESS
SERVICE (APS)**

 NGUEMNIN ALEX

 KAMDEM Jordan

Encadré par : **M. NJIKI Yannick**

ANNÉE ACADÉMIQUE

2024-2025

Table des matières

1. Constat(s) problématique(s) et sources	3
2. Historique des mesures déjà appliquées	3
3. Problèmes posés / écarts aux normes, objectifs, attentes ou enjeux	3
4. Indicateurs d'analyse et causes profondes	4
5. Analyse causale détaillée	5
6. Évaluation de l'étendue du (des) Problème(s) posé(s) (impacts chiffrés, parties prenantes)....	6
7. Positionnement dans la littérature et benchmarks	6
8. Recherches documentaires / experts	7
9. Indicateurs de suivi et mesure des effets	8
10. Solutions techniques + critères d'efficacité	9
11. Risques inhérents à la solution retenue et mesures de mitigation	10
12. Solutions de conduite du changement + critères d'efficacité	11
13. Personnes et ressources critiques impliquées	12
14. Évaluation coûts / bénéfices	13
15. Plan d'action technique (responsable, délai, livrables, critères d'atteintes)	14
16. Plan d'action communication et conduite du changement (responsable, délai, livrables)....	15
17. Avis du Comité CRA DSIT	17
18. Tableau de bord de suivi	17

Ce rapport détaille les spécifications révisées pour le système de monitoring Alfresco Process Services (APS), intégrant des modifications clés basées sur les retours opérationnels et les orientations stratégiques. L'objectif est de fournir une vue d'ensemble claire et précise des fonctionnalités, de l'architecture technique et des plans de mise en œuvre, en mettant l'accent sur la détection proactive des processus problématiques et l'optimisation des flux de travail.

1. Constat(s) problématique(s) et sources

Dans le secteur bancaire moderne, des institutions telles qu'Afriland First Bank s'appuient de plus en plus sur l'automatisation des processus métier pour améliorer leur efficacité opérationnelle, optimiser la gestion des risques et répondre aux attentes croissantes des clients en matière de services rapides et fiables. Alfresco Process Services (APS) s'est imposé comme un outil essentiel pour l'automatisation de divers flux de travail, incluant la gestion des crédits, l'approbation des prêts et la conformité réglementaire.

Cependant, malgré les avantages significatifs d'APS, des difficultés peuvent survenir, notamment en ce qui concerne les performances et la réactivité du système. Une lenteur notable est fréquemment observée, souvent attribuable à l'accumulation de processus abandonnés. Les problèmes identifiés incluent l'accumulation de données dans la base de données, les processus bloqués dès la première tâche utilisateur, les processus interrompus par des exceptions, une lenteur générale des performances, ainsi que des tâches non traitées et des utilisateurs inactifs. Ces problématiques constituent les moteurs principaux de la nécessité d'un système de monitoring dédié, visant à résoudre ces défis et à garantir une performance et une fiabilité optimales d'APS dans un environnement bancaire critique.

Les problèmes listés, tels que les processus bloqués, les exceptions système et les utilisateurs inactifs, sont des indicateurs directs de processus qui sont soit abandonnés, soit erronés. La présence de ces dysfonctionnements justifie pleinement l'orientation stratégique de la surveillance en temps réel vers la détection ciblée de ces états critiques. En se concentrant sur ces catégories spécifiques de problèmes, le système de monitoring pourra diriger l'attention des opérateurs vers les interventions les plus urgentes et les plus impactantes, optimisant ainsi l'efficacité de la gestion opérationnelle.

2. Historique des mesures déjà appliquées

Le document ne détaille pas les mesures spécifiques déjà appliquées pour résoudre les problématiques de performance et de fiabilité d'Alfresco Process Services (APS) avant la proposition de ce système de monitoring. Le projet actuel vise à mettre en place une nouvelle solution intégrée pour résoudre ces défis de manière exhaustive.

3. Problèmes posés / écarts aux normes, objectifs, attentes ou enjeux

Le système de monitoring proposé est conçu pour répondre aux besoins critiques identifiés afin d'optimiser la performance et la fiabilité d'Alfresco Process Services, transformant ainsi les défis actuels en opportunités d'amélioration opérationnelle. Les écarts par rapport aux objectifs de performance et de fiabilité se manifestent par plusieurs points essentiels :

- **Détection et Gestion Proactive des Processus Problématiques :** Le système actuel ne parvient pas à identifier automatiquement les processus bloqués, en particulier ceux qui stagnent dès la première tâche utilisateur ou qui dépassent des seuils de durée définis (par exemple, plus de X heures/jours). Il est impératif que ces processus soient visuellement marqués pour permettre une intervention rapide et ciblée. De plus, la capacité à afficher spécifiquement les processus susceptibles d'être abandonnés ou erronés fait défaut, entravant une réponse opérationnelle focalisée. Cette exigence marque un changement fondamental, passant d'une simple

vue d'ensemble à un système d'alerte précoce intelligent, qui dirige l'attention des opérateurs vers les problèmes les plus critiques plutôt que de présenter une quantité exhaustive de données.

- **Goulots d'étranglement dans les tâches et la performance utilisateur :** Il existe un manque de suivi détaillé des tâches en attente, assignées et en retard, ainsi qu'une incapacité à mesurer la performance individuelle des utilisateurs et des rôles (temps moyen de traitement) pour détecter les utilisateurs inactifs ou surchargés.
- **Absence de mécanismes d'alerte et de notification automatisés :** Le système actuel est incapable d'envoyer des alertes en temps réel lorsque des seuils critiques sont atteints (dépassement du nombre maximal de processus, tâches non assignées, augmentation des erreurs d'API ou de base de données).
- **Limitations des rapports et tableaux de bord :** Il est difficile d'obtenir une vue synthétique de la santé d'APS et de générer des rapports personnalisés exportables pour l'optimisation continue et l'aide à la décision stratégique.
- **Manque de fonctionnalités de maintenance automatisée :** L'absence de mécanismes pour archiver ou terminer les processus inactifs ou abandonnés entraîne un encombrement de la base de données et une dégradation des performances.

Ces problèmes compromettent l'efficacité opérationnelle, la gestion des risques et la capacité à répondre aux attentes des clients en matière de services rapides et fiables dans un environnement bancaire critique. La réorientation de la visibilité en temps réel vers la détection proactive des processus problématiques positionne le système de monitoring comme un outil d'aide à la décision et d'intervention rapide, plutôt qu'un simple tableau de bord de statut général.

4. Indicateurs d'analyse et causes profondes

Les indicateurs d'analyse et les causes profondes des problèmes identifiés dans Alfresco Process Services sont les suivants :

Indicateurs d'Analyse :

- **Statut des processus :** Nombre de processus actifs, terminés, suspendus et échoués. Bien que le système se concentre sur les processus problématiques, le suivi de ces statuts généraux reste pertinent pour une analyse contextuelle.
- **Durée des processus :** Temps d'exécution des processus, avec des seuils d'alerte configurables pour les processus anormalement longs.
- **Tâches :** Suivi détaillé des tâches en attente, assignées et en retard.
- **Performance utilisateur :** Temps moyen de traitement des tâches par utilisateur ou par rôle.
- **Erreurs système :** Augmentation des erreurs d'API ou de base de données, exceptions système.
- **Volume de processus :** Dépassement de la capacité maximale de processus.
- **Tendances historiques :** Échecs mensuels de processus, évolution du nombre de processus abandonnés.
- **Disponibilité du système :** Indicateur de la santé globale d'APS.

Causes Profondes :

- **Accumulation de processus abandonnés :** Cause principale de la lenteur notable et de l'encombrement de la base de données.
- **Processus bloqués à la première tâche utilisateur :** Souvent un indicateur clé de problèmes d'assignation ou de traitement initial.
- **Processus monopolisés par leur initiateur :** Entraîne des blocages et une dépendance excessive à une seule personne.

- **Exceptions système** : Échecs d'API ou erreurs de base de données qui interrompent les processus.
- **Saturation de la base de données** : Résultat de l'accumulation de données dans les tables ACT_RU_VARIABLE et ACT_HI_VARINST.
- **Goulots d'étranglement** : Dans l'assignation ou le traitement des tâches, ou liés à des utilisateurs inactifs ou surchargés.

Les causes profondes identifiées, telles que l'accumulation de processus abandonnés et les processus bloqués dès la première tâche utilisateur, constituent la justification fondamentale de l'orientation de la visibilité en temps réel vers les processus "susceptibles d'être abandonnés ou erronés". Les indicateurs du système sont conçus pour directement mesurer et alerter sur ces états problématiques, permettant une intervention ciblée et efficace.

5. Analyse causale détaillée

L'analyse détaillée des causes profondes révèle plusieurs facteurs interdépendants contribuant aux problèmes de performance et de fiabilité d'Alfresco Process Services (APS) :

- **Accumulation de processus abandonnés et inactifs** : La cause fondamentale de la lenteur du système est l'accumulation de processus qui ne sont pas terminés ou archivés. Ces processus, qu'ils soient réellement abandonnés ou simplement inactifs, continuent de consommer des ressources et d'encombrer la base de données, notamment les tables ACT_RU_VARIABLE et ACT_HI_VARINST. Cela conduit à une dégradation progressive des performances générales d'APS.
- **Processus bloqués à la première tâche utilisateur** : Un problème récurrent est le blocage des processus dès leur première tâche utilisateur. Cela est souvent dû à des problèmes d'assignation (la tâche n'est pas attribuée ou l'utilisateur assigné est inactif) ou à un manque de traitement initial, ce qui empêche le flux de travail de progresser.
- **Processus monopolisés par leur initiateur** : Certains processus restent bloqués parce qu'ils sont exclusivement dépendants de leur initiateur, qui peut être inactif ou surchargé. Cette dépendance crée des points de défaillance uniques et ralentit l'ensemble du système.
- **Erreurs et exceptions système** : Les échecs d'API ou les erreurs de base de données peuvent interrompre les processus de manière inattendue. Sans un mécanisme de suivi et de reprise adéquat, ces interruptions entraînent des processus incomplets et nécessitent des interventions manuelles coûteuses.
- **Manque de visibilité et de gestion proactive** : L'absence d'un système de monitoring en temps réel empêche l'identification rapide des goulots d'étranglement, des processus bloqués ou des erreurs. Les équipes ne peuvent pas réagir de manière proactive, ce qui transforme les petits problèmes en blocages majeurs et en dégradations de performance.
- **Absence de maintenance automatisée** : Le manque de mécanismes automatiques pour archiver ou terminer les processus obsolètes ou inactifs aggrave l'encombrement de la base de données et la lenteur du système, créant un cercle vicieux de dégradation des performances.

Ces causes combinées créent un environnement où l'efficacité opérationnelle est compromise, les risques augmentent et la réactivité du système diminue, impactant directement les services bancaires critiques. La compréhension approfondie de ces causes structurelles justifie l'orientation du système de monitoring vers la détection et la gestion proactive des processus problématiques, car cibler ces points de défaillance fondamentaux est la voie la plus efficace pour restaurer la performance et la fiabilité d'APS.

6. Évaluation de l'étendue du (des) Problème(s) posé(s) (impacts chiffrés, parties prenantes)

Les problèmes identifiés dans Alfresco Process Services (APS) ont des impacts significatifs sur l'institution bancaire, **bien que des chiffres précis ne soient pas fournis dans le document.**

Impacts :

- **Lenteur opérationnelle :** La "lenteur notable" du système se traduit par des délais accrus dans le traitement des opérations bancaires (gestion des crédits, approbation des prêts, conformité réglementaire), ce qui peut affecter la satisfaction client et la compétitivité de la banque.
- **Dégradation des performances de la base de données :** L'accumulation de processus abandonnés et de données inutiles entraîne un "encombrement de la base de données", ce qui ralentit les requêtes et les opérations, augmentant potentiellement les coûts d'infrastructure et de maintenance.
- **Augmentation des interventions manuelles :** Les processus bloqués, les erreurs non gérées et les tâches non traitées nécessitent des "interventions manuelles coûteuses en temps et en effort de débogage", détournant les ressources humaines de tâches à plus forte valeur ajoutée.
- **Risques opérationnels accrus :** Le manque de visibilité et de réactivité face aux processus problématiques peut entraîner des retards dans la conformité réglementaire, des erreurs dans la gestion des risques et une exposition accrue aux incidents opérationnels.
- **Impact sur l'expérience utilisateur :** La lenteur et les blocages affectent directement l'expérience des utilisateurs d'APS (employés de la banque), réduisant leur productivité et pouvant générer de la frustration.

Parties Prenantes Impactées :

- **Afriland First Bank :** L'institution dans son ensemble est affectée par la baisse d'efficacité opérationnelle, l'augmentation des risques et l'impact sur la satisfaction client.
- **Administrateurs (Managers) :** Ils sont directement responsables de la résolution des problèmes détectés par le système (processus bloqués, erreurs critiques) et sont chargés d'intervenir et de résoudre la situation.
- **Responsables de groupe :** Ils doivent pouvoir consulter et agir sur les processus de leur groupe attribué.
- **Utilisateurs d'APS :** Les employés qui interagissent avec les processus métier sont impactés par la lenteur et les blocages, ce qui réduit leur productivité.

Bien que des chiffres précis sur les pertes financières ou les retards ne soient pas disponibles, les impacts qualitatifs décrits soulignent une dégradation significative de l'efficacité et de la fiabilité des opérations bancaires. La responsabilité de produire un code frontend de haute qualité, réactif et maintenable incombe entièrement aux équipes techniques compétentes. Cela met en lumière l'importance de workflows de conception-développement robustes et de standards de codage rigoureux pour éviter l'accumulation de dette technique.

7. Positionnement dans la littérature et benchmarks

Dans le contexte actuel du développement logiciel, les organisations, en particulier dans le secteur bancaire, sont confrontées à la décision stratégique d'adopter des solutions tierces ou de capitaliser sur leurs capacités de développement internes. Pour le système de monitoring APS, la banque a opté pour une approche de développement sur mesure, s'appuyant sur ses équipes techniques adéquates et compétentes.

Cette orientation stratégique est motivée par plusieurs considérations. Premièrement, le développement interne offre un contrôle total sur l'architecture, la personnalisation et l'intégration du système, ce qui est crucial pour des applications bancaires critiques nécessitant une conformité réglementaire stricte et des exigences de sécurité élevées. Les équipes internes peuvent adapter précisément le système aux besoins spécifiques d'Afriland First Bank, garantissant une intégration transparente avec l'écosystème SI existant.

Deuxièmement, bien que les outils d'accélération puissent promettre des gains de temps initiaux, le développement sur mesure par des équipes expérimentées permet une meilleure maîtrise de la qualité du code, de sa sémantique et de sa maintenabilité à long terme. Cela réduit la dépendance vis-à-vis des technologies tierces et minimise les risques liés à la dette technique potentiellement introduite par le code généré automatiquement, qui pourrait ne pas toujours adhérer aux standards internes ou aux exigences de performance spécifiques.

Enfin, cette approche favorise le renforcement des compétences internes et la capitalisation sur l'expertise technique de la banque. En investissant dans ses propres équipes, Afriland First Bank développe une capacité durable à innover, à maintenir et à faire évoluer ses systèmes critiques de manière autonome, assurant ainsi une plus grande agilité et résilience face aux évolutions technologiques et aux besoins métier futurs. Cette stratégie met en avant la valeur intrinsèque de l'ingénierie logicielle humaine et de la collaboration directe entre les équipes de conception et de développement pour des résultats optimaux.

8. Recherches documentaires / experts

La solution proposée pour le système de monitoring APS s'appuie sur une approche de développement robuste et éprouvée, guidée par des principes d'ingénierie logicielle solides et l'expertise des équipes techniques internes. Cette section détaille les fondements méthodologiques et techniques qui sous-tendent la conception et la mise en œuvre du système.

Le développement d'une application réactive en temps réel, comme le système de monitoring APS, nécessite une compréhension approfondie des architectures orientées services et des paradigmes de programmation non-bloquants. L'adoption de Spring WebFlux pour le backend et de WebSockets pour la communication frontend-backend est une décision technique stratégique qui s'inscrit dans les meilleures pratiques pour les systèmes nécessitant une haute réactivité et un traitement efficace des flux de données continus. Spring WebFlux, en tant que framework réactif, permet de gérer un grand nombre de connexions simultanées avec une utilisation optimisée des ressources, ce qui est essentiel pour un monitoring en temps réel des processus APS. Les WebSockets, quant à eux, établissent une connexion bidirectionnelle persistante, garantissant que les mises à jour sur les processus problématiques sont poussées instantanément vers l'interface utilisateur, sans les délais inhérents aux requêtes HTTP traditionnelles.

La conception de l'interface utilisateur, réalisée avec Figma, est le point de départ du processus de développement frontend. Les designers UI/UX se concentrent sur la création de maquettes ergonomiques et visuellement fidèles à la charte graphique de la banque (rouge, noir, blanc). La qualité de ces designs, incluant l'utilisation rigoureuse de l'Auto Layout et une structuration logique des calques, est primordiale pour faciliter l'implémentation par les développeurs frontend.

En somme, la recherche et l'expertise se concentrent sur l'application de principes d'ingénierie logicielle éprouvés, la sélection de technologies adaptées aux exigences de réactivité et la valorisation du savoir-faire des équipes internes pour construire un système de monitoring fiable, performant et évolutif.

9. Indicateurs de suivi et mesure des effets

Le système de monitoring d'Alfresco Process Services (APS) est conçu pour fournir des indicateurs clés permettant un suivi précis et une mesure des effets des améliorations apportées. Ces indicateurs sont essentiels pour l'optimisation continue et l'aide à la décision stratégique. La visibilité ciblée sur les processus problématiques est au cœur de cette approche.

Indicateurs de Suivi :

- **Visibilité en Temps Réel Ciblée :**
 - Identification et décompte des processus *susceptibles d'être abandonnés ou erronés*. Cette focalisation permet aux opérateurs de diriger immédiatement leur attention vers les points de défaillance critiques, réduisant ainsi la surcharge d'informations et optimisant les temps de réponse.
 - Processus bloqués : Identification des processus stagnants au-delà d'une durée prédéfinie (par exemple, plus de X heures/jours), avec une attention particulière aux blocages dès la première tâche utilisateur. Ces indicateurs sont directement liés aux causes profondes de la lenteur du système.
 - Temps d'exécution des processus problématiques : Surveillance continue du temps de vie des processus qui dépassent les seuils définis, avec des alertes configurables en cas de durée anormalement longue.
- **Gestion des Tâches et Performance Utilisateur :**
 - Suivi détaillé des tâches en attente, assignées et en retard.
 - Temps moyen de traitement des tâches par utilisateur ou par rôle.
 - Détection des utilisateurs inactifs ou surchargés.
- **Suivi des Erreurs et Exceptions :**
 - Journalisation et classification des exceptions système (échecs d'API, erreurs de base de données), crucial pour identifier les processus "erronés".
- **Alertes et Notifications :**
 - Nombre d'alertes déclenchées pour le dépassement de la capacité maximale de processus, les tâches non assignées depuis un certain temps, ou l'augmentation des erreurs d'API ou de base de données.
 - Efficacité des notifications multi-canaux (email, Teams, SMS).
- **Rapports et Tableaux de Bord :**
 - Disponibilité du système APS.
 - Nombre d'utilisateurs actifs.
 - Volume de processus traités.
 - Analyse du cycle de vie des processus (durée de traitement moyenne).
 - Tendances historiques : Échecs mensuels de processus, évolution du nombre de processus abandonnés.

Mesure des Effets :

La mesure des effets se fera par l'analyse des tendances de ces indicateurs au fil du temps. Par exemple, une réduction du nombre de processus abandonnés, une diminution du temps moyen de traitement des tâches, une meilleure répartition de la charge de travail entre les utilisateurs, et une diminution des erreurs système indiqueront l'efficacité du système de monitoring. La capacité à générer des rapports personnalisés exportables (CSV/PDF) permettra une analyse approfondie et une aide à la décision stratégique pour l'optimisation continue. Cette approche axée sur la détection proactive des problèmes permet une meilleure allocation des ressources humaines et une intervention plus rapide, renforçant ainsi la résilience opérationnelle de la banque.

10. Solutions techniques + critères d'efficacité

Le système de monitoring pour Alfresco Process Services (APS) est conçu comme une application web orientée services, avec une architecture monolithique, utilisant des technologies spécifiques pour le backend, le frontend et la base de données.

10.1. Solutions Techniques du Système de Monitoring :

- **Architecture** : L'application sera développée selon une architecture **monolithique**. Ce choix architectural est retenu pour garantir une gestion simplifiée du déploiement et une cohérence accrue du code, tout en permettant une évolutivité et une facilité de maintenance pour le périmètre défini.
- **Backend** : La couche backend sera impérativement développée avec le framework JAVA (Spring Boot). Pour répondre aux exigences de réactivité et de traitement des flux de données en temps réel, le backend exploitera **Spring WebFlux**. Ce framework permet une programmation réactive et non-bloquante, essentielle pour gérer efficacement les volumes de données de monitoring et les mises à jour instantanées de l'état des processus. Ce choix assure une base solide et performante pour la logique métier du système de monitoring.
- **Frontend** : La couche de présentation sera développée avec le framework Angular 19, en respectant rigoureusement une architecture MVC (Model-View-Controller). L'intégration des **WebSockets** sera impérative pour garantir une visibilité en temps réel instantanée et des mises à jour push des processus problématiques. Cette technologie permet une communication bidirectionnelle persistante entre le serveur et le client, assurant que les informations critiques sur les processus abandonnés ou erronés sont affichées sans délai.
- **Base de Données** : La base de données utilisée pour le stockage des données du système de monitoring sera PostgreSQL. Ce choix offre robustesse, fiabilité et performance pour la gestion des métriques et des historiques de monitoring.

Fonctionnalités Clés :

- **Gestion des Utilisateurs et Rôles** : Définition de rôles (Administrateur/Manager, Responsable de groupe) avec des permissions claires pour le contrôle d'accès et l'authentification. Les processus pourront être attribués à des groupes ou à une personne particulière, offrant une granularité accrue dans la gestion des assignations et des responsabilités.
- **Surveillance et Détection Proactive des Processus Problématiques** : Le système assurera le suivi des processus actifs et la détection des processus bloqués, y compris ceux qui stagnent dès la première tâche utilisateur ou qui dépassent des seuils de durée prédéfinis. Le système affichera spécifiquement les processus susceptibles d'être abandonnés ou erronés, dirigeant ainsi l'attention des opérateurs vers les points d'intervention critiques.
- **Gestion des Tâches et Analyse des Workflows** : Surveillance des tâches en attente, assignées et en retard, ainsi que la mesure de la performance des utilisateurs. La réaffectation des tâches sera effectuée manuellement par les administrateurs. Cette approche privilégie le contrôle humain pour les décisions complexes de réaffectation, reconnaissant la nécessité d'une intervention délibérée dans des flux de travail bancaires critiques.
- **Suivi des Erreurs et Exceptions** : Journalisation et classification des exceptions système (échecs d'API, erreurs de base de données).
- **Alertes et Notifications** : Envoi d'alertes basées sur des seuils critiques via des notifications multi-canaux (email, Teams, SMS).
- **Rapports et Tableaux de Bord** : Tableaux de bord prédéfinis et rapports personnalisés exportables (CSV/PDF) pour l'analyse et l'aide à la décision, axés sur les processus problématiques.
- **Actions de Maintenance Automatisées** : Nettoyage des processus abandonnés (archivage/termination), mécanismes de reprise des appels d'API échoués, et système de notification automatique pour les processus nécessitant une action.

10.2. Développement Frontend :

Le développement de l'interface utilisateur sera réalisé par des équipes techniques adéquates et compétentes, en utilisant le framework Angular 19. L'accent sera mis sur la création d'une interface utilisateur moderne, réactive et ergonomique, garantissant une fidélité visuelle aux designs approuvés et une expérience utilisateur optimale. La conception des composants UI sera basée sur les meilleures pratiques de développement frontend pour assurer la modularité, la réutilisabilité et la maintenabilité du code. Cette approche garantit un contrôle total sur la qualité et la performance du code, en ligne avec les standards élevés requis pour une application bancaire.

10.3. Critères d'Efficacité des Solutions :

- **Performance et Fiabilité** : Le système doit garantir une performance et une fiabilité optimales d'APS, notamment en réduisant la lenteur et l'encombrement de la base de données.
- **Visibilité Ciblée et Proactivité** : Capacité à offrir une visibilité immédiate *exclusivement* sur l'état des processus problématiques (abandonnés, erronés) et à détecter proactivement ces problèmes pour une intervention rapide.
- **Réduction des Interventions Manuelles** : Minimisation des interventions humaines grâce aux mécanismes de reprise et aux alertes ciblées, bien que la réaffectation des tâches reste manuelle. Cette distinction est importante, car le système réduit l'effort de détection tout en maintenant la supervision humaine pour la résolution.
- **Qualité du Code** : Le code produit par les équipes techniques doit être "developer-friendly", sémantique, bien commenté et structuré logiquement, facilitant sa compréhension et son édition.
- **Efficacité du Développement** : Le développement sera mené par des équipes techniques compétentes, en se concentrant sur la production d'un code robuste et maintenable, avec des délais définis par les méthodologies de projet classiques. La productivité sera mesurée par la capacité de l'équipe à livrer des fonctionnalités de haute qualité dans les délais impartis.
- **Réactivité** : L'interface utilisateur doit être intrinsèquement réactive et s'adapter aux différentes tailles d'écran, grâce à l'utilisation de WebSockets et des meilleures pratiques Angular.
- **Maintenabilité et Évolutivité** : L'architecture monolithique et le code modulaire doivent garantir une application facile à maintenir et à faire évoluer.

11. Risques inhérents à la solution retenue et mesures de mitigation

L'adoption d'une approche de développement sur mesure pour le système de monitoring APS, bien qu'offrant un contrôle accru et une personnalisation profonde, présente certains risques qui nécessitent des mesures de mitigation proactives.

11.1. Risques Inhérents à la Solution Retenue (Développement Custom) :

- **Complexité de Développement** : Le développement d'une application réactive en temps réel, utilisant des technologies comme Spring WebFlux et WebSockets, et s'inscrivant dans une architecture monolithique robuste, peut présenter des défis de complexité significatifs. Cela inclut la gestion des états, la propagation des erreurs dans un environnement non-bloquant et l'optimisation des performances sous forte charge de données.
- **Qualité et Cohérence du Code** : Il existe un risque d'accumulation de dette technique si les standards de codage, les revues de code et les pratiques de test ne sont pas rigoureusement appliqués.
- **Délais de Développement** : Le développement de l'ensemble de l'application peut entraîner des délais plus. Cela peut impacter le temps de mise sur le marché des fonctionnalités et la réactivité aux besoins métier urgents.
- **Sécurité** : La sécurité de l'application, tant au niveau du backend que du frontend, doit être assurée par des pratiques de codage sécurisées, des tests de pénétration réguliers et des revues de sécurité approfondies.

- **Performance** : Bien que les technologies choisies soient adaptées à la réactivité, assurer la performance et la scalabilité de l'application monolithique sous une charge élevée de données en temps réel nécessite une conception minutieuse et des optimisations continues.
- **Erreur Humaine dans la Réaffectation** : La décision de réaffecter manuellement les tâches introduit un risque d'erreurs humaines, de retards dans la résolution des blocages de processus, ou d'incohérences si les procédures ne sont pas claires ou si le personnel n'est pas suffisamment formé.

11.2. Mesures de Mitigation :

- **Adoption de Bonnes Pratiques de Développement** : Il est impératif d'implémenter rigoureusement les standards de codage, de mettre en place des revues de code régulières et d'effectuer des tests unitaires, d'intégration, système et de performance approfondis. Cela garantit la robustesse et la qualité du code produit.
- **Expertise Technique** : S'assurer que les équipes techniques adéquates et compétentes possèdent l'expertise nécessaire en Spring WebFlux, WebSockets, Angular 19, et PostgreSQL. Des formations continues et le recrutement de profils spécialisés peuvent être nécessaires.
- **Gestion de Projet Agile** : L'utilisation de méthodologies agiles permettra une livraison incrémentale des fonctionnalités, une adaptation rapide aux changements de spécifications et une gestion proactive des risques tout au long du cycle de vie du projet.
- **Tests de Charge et de Performance** : La réalisation de tests de charge et de stress est essentielle pour valider la capacité du système à gérer le volume de données en temps réel et à maintenir ses performances sous des conditions de charge extrêmes.
- **Audits de Sécurité** : Conduire des audits de sécurité réguliers et intégrer la sécurité dès la conception (Security by Design) pour identifier et corriger les vulnérabilités potentielles avant le déploiement.
- **Documentation Rigoureuse** : Maintenir une documentation technique et fonctionnelle complète et à jour pour faciliter la maintenance, l'évolution et l'onboarding des nouvelles ressources.

Ces mesures permettent de gérer efficacement les risques associés au développement sur mesure, assurant ainsi la production d'une application de monitoring de haute qualité, sécurisée et performante.

12. Solutions de conduite du changement + critères d'efficacité

La mise en œuvre du système de monitoring APS et l'adoption des nouvelles méthodologies de développement nécessitent une stratégie de conduite du changement pour assurer une adoption réussie et maximiser les bénéfices.

12.1. Solutions de Conduite du Changement :

- **Formation et Accompagnement des Utilisateurs du Système de Monitoring** :
 - **Cible** : Administrateurs (Managers), Responsables de groupe, et tout utilisateur d'APS concerné par le suivi des processus.
 - **Contenu** : Sessions de formation interactives sur l'utilisation des tableaux de bord (axés sur les processus problématiques), l'interprétation des indicateurs, la configuration des alertes, le déclenchement des actions de maintenance (archivage), et l'exportation des rapports.
 - **Support** : Mise en place d'un support continu (documentation, FAQ, points de contact dédiés) pour répondre aux questions et résoudre les difficultés initiales.
- **Sensibilisation et Communication sur les Bénéfices** :
 - **Cible** : Toutes les parties prenantes, les équipes opérationnelles et les développeurs.

- **Message** : Mettre en avant les gains d'efficacité opérationnelle, la réduction des risques grâce à la détection proactive des problèmes, l'amélioration de l'expérience utilisateur, et la scalabilité du système d'automatisation des processus métier.
- **Canaux** : Réunions d'information, communications internes (emails, intranet), démonstrations du système.
- **Adoption des Nouvelles Méthodologies de Développement et Collaboration** :
 - **Cible** : Équipes de design et de développement.
 - **Formation** : Ateliers pratiques sur les meilleures pratiques de conception UI/UX en Figma pour un développement frontend manuel par des équipes techniques compétentes.
 - **Culture de Collaboration** : Promouvoir une culture de collaboration étroite et directe entre les designers et les développeurs, axée sur la communication continue et l'itération rapide. Cela favorisera une meilleure compréhension mutuelle des contraintes et des objectifs.
 - **Intégration des Outils** : Encourager l'utilisation des outils de développement et de gestion de version standard (ex: GitHub, VS Code) pour une synchronisation continue et une gestion fluide des conflits de code.

12.2. Critères d'Efficacité de la Conduite du Changement :

- **Taux d'Adoption** : Pourcentage d'utilisateurs qui utilisent activement le système de monitoring et les nouvelles méthodologies de développement dans leur travail quotidien.
- **Satisfaction Utilisateur** : Mesure de la satisfaction des utilisateurs vis-à-vis du nouveau système et des outils, via des enquêtes ou des retours qualitatifs.
- **Réduction des Demandes de Support** : Diminution du nombre de questions ou de problèmes liés à l'utilisation du système après la période initiale de formation.
- **Amélioration de la Collaboration** : Évaluation de la fluidité de la collaboration entre designers et développeurs, mesurée par la réduction des allers-retours.
- **Impact sur la Productivité du Développement** : Mesure de la productivité des équipes techniques compétentes dans la livraison de fonctionnalités, en se basant sur les estimations de projet et les délais réels.
- **Atteinte des Objectifs du Système de Monitoring** : Le système doit effectivement permettre une gestion proactive des défis liés aux processus abandonnés et erronés, renforçant l'efficacité opérationnelle et la gestion des risques.

En mettant en œuvre ces solutions de conduite du changement, l'organisation s'assurera que les équipes sont non seulement équipées des bons outils, mais aussi qu'elles les adoptent pleinement, maximisant ainsi le retour sur investissement du projet. La formation sur la réaffectation manuelle des tâches est essentielle pour garantir que les opérateurs sont pleinement préparés à leurs nouvelles responsabilités opérationnelles.

13. Personnes et ressources critiques impliquées

La réussite de ce projet repose sur l'implication de plusieurs personnes clés et l'utilisation de ressources techniques spécifiques. L'approche de développement sur mesure met un accent particulier sur l'expertise humaine et les outils fondamentaux de l'ingénierie logicielle.

13.1. Personnes clés Impliquées :

- **M. NJIKI Yannick** : Contact principal et ressource désignée pour le projet (yannick_njiki@afriandfirstbank.com). Joue un rôle de supervision et de conseil technique.
- **Administrateur (Manager)** : Responsable de la gestion complète du système de monitoring, de la résolution des problèmes APS, de la configuration des seuils d'alerte, de la gestion des utilisateurs du monitoring, et de l'initiation des actions de maintenance.
- **Responsable de Groupe** : Doit pouvoir consulter et agir uniquement sur les processus de son groupe attribué.

- **Équipe de Développement :**
 - **Développeurs Backend :** Spécialisés en JAVA (Spring Boot), avec une expertise particulière en **Spring WebFlux** pour la logique métier réactive du système de monitoring.
 - **Développeurs Frontend :** Spécialisés en Angular 19 et maîtrisant l'intégration des **WebSockets** pour la couche de présentation et la communication en temps réel. Ils seront les principaux artisans de la traduction des designs UI/UX en code fonctionnel et réactif.
- **Designers UI/UX :** Responsables de la conception des interfaces dans Figma, en respectant la charte graphique d'Afriland (rouge, noir, blanc) et en optimisant les designs pour une implémentation efficace et de haute qualité par les équipes techniques compétentes.

13.2. Ressources Techniques Critiques :

- **Outils de Design :**
 - **Figma :** Outil de design principal pour la création des interfaces utilisateur.
- **Environnement de Développement :**
 - **Visual Studio Code (VS Code) :** IDE recommandé pour le développement.
 - **GitHub :** Dépôt de code pour la gestion des versions et la collaboration continue.
- **Technologies de Développement :**
 - **Backend :** JAVA (Spring Boot, **Spring WebFlux**).
 - **Frontend :** Angular 19, **WebSockets**.
 - **Base de Données :** PostgreSQL.

Ces personnes et ressources sont interdépendantes. Leur coordination efficace et la valorisation de l'expertise interne sont cruciales pour le succès du projet de mise en place du système de monitoring, en particulier avec le choix d'un développement sur mesure.

14. Évaluation coûts / bénéfices

L'évaluation des coûts et bénéfices du projet de mise en place d'un système de monitoring pour APS, s'appuyant sur des équipes techniques compétentes, se base sur les informations disponibles, bien que des chiffres monétaires précis ne soient pas fournis.

14.1. Bénéfices :

- **Efficacité du Développement Frontend :** Le développement sera mené par des équipes techniques adéquates et compétentes, garantissant un code robuste, maintenable et aligné avec les standards internes. Cela permet aux développeurs de se concentrer sur la logique métier complexe et les défis techniques, assurant une solution de haute qualité.
- **Qualité et Maintenabilité du Code :** Le code produit par les équipes techniques sera intrinsèquement réactif, modulaire et hautement extensible. Il sera propre, sémantique, correctement commenté et structuré, facilitant la maintenance et les futures évolutions.
- **Amélioration de l'Efficacité Opérationnelle d'APS :** Le système permettra une optimisation de la performance et de la fiabilité d'Alfresco Process Services. Il contribuera à la réduction de l'encombrement de la base de données et à l'amélioration des performances grâce au nettoyage des processus abandonnés. L'augmentation de la résilience du système et la minimisation des interventions manuelles seront assurées par les mécanismes de reprise et les alertes ciblées. La visibilité ciblée en temps réel et la détection proactive des problèmes permettront des interventions rapides et efficaces.
- **Collaboration Améliorée :** Des processus de travail clairs et une communication directe entre designers et développeurs favoriseront une collaboration fluide et efficace, réduisant les frictions et les allers-retours. Cela permettra de maintenir la synchronisation entre le design et le code tout au long du cycle de vie du projet.
- **Expérience Utilisateur Améliorée :** Pour les utilisateurs d'APS, une meilleure réactivité et moins de blocages se traduiront par une productivité accrue et une meilleure expérience. Pour

les utilisateurs du système de monitoring, une interface ergonomique et des tableaux de bord clairs faciliteront la prise de décision.

- **Gestion des Risques Optimisée :** Le système de monitoring permettra une gestion proactive des défis liés aux processus abandonnés et erronés, positionnant la banque pour une croissance durable et une gestion des risques optimisée.

14.2. Coûts :

Bien que des chiffres monétaires précis ne soient pas fournis, les coûts peuvent inclure :

- **Ressources Humaines :** Coûts liés aux salaires des développeurs (backend, frontend), designers, chefs de projet, et personnel de support.
- **Infrastructure :** Coûts des serveurs, bases de données (PostgreSQL), et autres services cloud nécessaires au déploiement de l'application de monitoring.
- **Formation :** Coûts associés à la formation des équipes à l'utilisation du nouveau système et aux méthodologies de développement spécifiques.
- **Gestion des Risques :** Coûts potentiels liés à la mitigation des risques (ex: revue de code pour la sécurité, tests approfondis).

En résumé, les bénéfices attendus, notamment l'amélioration de la qualité du code, l'optimisation des opérations APS et le renforcement des compétences internes, semblent largement justifier les investissements nécessaires en ressources humaines et en temps de développement. La décision de privilégier le développement sur mesure par des équipes compétentes reflète une volonté de garantir une solution stratégiquement alignée et durable.

15. Plan d'action technique (responsable, délai, livrables, critères d'atteintes)

Ce plan d'action technique détaille les étapes pour la mise en œuvre du système de monitoring APS, en s'appuyant sur les équipes techniques compétentes pour le développement.

Responsable Principal : Équipe de Développement (Backend et Frontend), sous la supervision de **M. NJIKI Yannick**

Délai : Non spécifié dans le document, à définir par l'équipe projet.

Phase	Re-sponsable(s)	Livrables	Critères d'Atteinte
1. Conception et Optimisation du Design Figma	Designers UI/UX	Designs Figma optimisés pour un développement frontend manuel par des équipes techniques compétentes (Auto Layout, Frames, nommage clair)	Designs conformes aux meilleures pratiques de design UI/UX pour un développement Angular/Ionic ; Fidélité visuelle à la charte graphique Afriland (rouge, noir, blanc).
2. Développement Frontend (Angular 19)	Développeurs Frontend	Code Angular développé manuellement (composants UI, réactivité, interactivité) ; Intégration des WebSockets pour la communication en temps réel	Code Angular produit par les équipes techniques ; Code propre, sémantique, structuré en composants ; Fonctionnalités interactives et réactives.

3. Développement Backend (Spring Boot avec Web-Flux)	Développeurs Backend	API REST pour la gestion des données APS et du monitoring ; Logique métier réactive pour la surveillance, les alertes et la maintenance ; Utilisation de Spring WebFlux	API fonctionnelles et sécurisées ; Respect de l'architecture monolithique ; Performances conformes aux attentes pour le traitement des flux de données en temps réel.
4. Mise en place de la Base de Données	Développeurs Backend	Schéma de base de données PostgreSQL ; Base de données configurée pour le stockage des métriques et historiques	Base de données robuste, fiable et performante ; Intégrité des données assurée.
5. Intégration Frontend-Backend	Développeurs Frontend & Backend	Connexion des composants Angular aux API Spring Boot ; Liaison de données (data binding) ; Communication temps réel via WebSockets	Flux de données bidirectionnel fonctionnel ; Affichage correct des données en temps réel.
6. Implémentation des Fonctionnalités de Monitoring	Développeurs Frontend & Backend	Modules de surveillance ciblée des processus problématiques, détection des processus bloqués, gestion des tâches, suivi des erreurs, alertes et notifications, rapports et tableaux de bord, actions de maintenance automatisées	Toutes les fonctionnalités listées dans la section 0 sont opérationnelles et répondent aux besoins exprimés, avec un accent sur la détection proactive des processus abandonnés ou erronés et la réaffectation manuelle des tâches.
7. Tests et Débogage	Équipe de Développement, QA	Rapports de tests (unitaires, intégration, système, performance) ; Correction des bugs	Stabilité et fiabilité du système ; Conformité aux spécifications fonctionnelles et non fonctionnelles ; Gestion des "safe area issues" sur mobile.
8. Déploiement	Développeurs Backend, DevOps	Application de monitoring déployée dans l'environnement de production	Accessibilité et disponibilité du système ; Intégration continue/déploiement continu (CI/CD) si mis en place.

Le développement frontend, représente une phase cruciale qui nécessite une planification détaillée et une exécution rigoureuse par les développeurs. La réussite de cette phase dépendra directement de la capacité des équipes à traduire les designs en code de haute qualité.

16. Plan d'action communication et conduite du changement (responsable, délai, livrables)

Ce plan vise à assurer une communication efficace et une adoption réussie du nouveau système de monitoring APS et des méthodologies de développement associées.

Responsable Principal : NJIKI Yannick pour la communication technique et stratégique.

Délai : Non spécifié dans le document, à définir par l'équipe projet.

Phase	Responsable(s)	Livrables	Critères d'Atteinte
1. Communication Initiale et Sensibilisation	Managers, M. NJIKI Yannick	Présentations aux équipes concernées (direction, opérations, IT) ; Communications internes (emails, intranet) sur les objectifs et bénéfices du projet (notamment la détection proactive des processus problématiques)	Compréhension claire des enjeux et de la valeur ajoutée du système ; Engagement des parties prenantes.
2. Formation des Utilisateurs Clés du Monitoring	Managers, M. NJIKI Yannick, Équipe de Développement	Sessions de formation interactives pour les administrateurs et responsables de groupe (axées sur l'utilisation des tableaux de bord pour les processus problématiques) ; Guides d'utilisation détaillés. Formation spécifique sur les procédures de réaffectation manuelle des tâches	Maîtrise de l'utilisation du système de monitoring ; Capacité à configurer et interpréter les données pour une intervention ciblée.
3. Formation aux Méthodologies de Développement et Collaboration	Designers UI/UX, Développeurs Frontend	Ateliers pratiques sur les meilleures pratiques Figma pour un développement manuel (Auto Layout, composants) ; Sessions sur l'intégration du code développé dans Angular/Ionic	Designs Figma optimisés pour le développement manuel ; Réduction des ajustements manuels post-conception ; Amélioration de la collaboration design-développement.
4. Mise en Place des Canaux de Notification	Développeurs Backend, Frontend	Configuration des alertes multi-canaux (email, Teams, SMS) ; Tests des notifications	Notifications "parviennent aux bonnes personnes au bon moment" pour les processus problématiques ; Fiabilité des alertes.
5. Démonstrations et Retours d'Expérience	Managers, M. NJIKI Yannick, Équipe de Développement	Démonstrations régulières des progrès et des fonctionnalités du système ; Sessions de feedback avec les utilisateurs finaux	Recueil des retours pour des ajustements continus ; Amélioration de l'expérience utilisateur.
6. Documentation et Support Continu	Équipe de Développement, Managers, M. NJIKI Yannick	Documentation technique et utilisateur à jour; FAQ ; Points de contact pour le support	Autonomie des utilisateurs ; Résolution rapide des problèmes.
7. Suivi de l'Adoption et des Bénéfices	Managers, M. NJIKI Yannick	Rapports d'adoption du système ; Évaluation des gains d'efficacité opérationnelle et de la réduction des problèmes APS	Taux d'adoption élevé ; Atteinte des objectifs d'amélioration de la performance et de la fiabilité d'APS (particulièrement sur la réduction des processus abandonnés ou erronés).

La formation sur les procédures de réaffectation manuelle des tâches est un élément critique de ce plan. Elle garantira que les utilisateurs finaux sont pleinement préparés à leurs responsabilités opérationnelles, compensant l'absence de mécanismes d'automatisation dans ce domaine.

17. Avis du Comité CRA DSIT

Cette section est réservée à l'avis et aux recommandations du Comité CRA DSIT, qui sera sollicité pour évaluer la pertinence et l'alignement du projet avec les stratégies et normes de l'organisation.

18. Tableau de bord de suivi

Le système de monitoring lui-même fournira un tableau de bord de suivi complet pour évaluer en temps réel et historiquement la santé et la performance d'Alfresco Process Services (APS). Ce tableau de bord est conçu pour offrir une visibilité ciblée et actionable.

Composants Clés du Tableau de Bord :

- **Vue d'Ensemble de la Santé d'APS (Ciblée) :**
 - Disponibilité du système.
 - Nombre d'utilisateurs actifs.
 - Volume de processus *susceptibles d'être abandonnés ou erronés*. Cette métrique clé reflète la nouvelle orientation du système, en concentrant l'attention sur les processus nécessitant une intervention.
 - Durée de traitement moyenne des processus *à risque*.
- **Performance des Processus :**
 - Graphiques des processus bloqués (par durée, par première tâche utilisateur).
 - Tendances des échecs de processus (mensuels, hebdomadaires).
 - Évolution du nombre de processus abandonnés.
- **Gestion des Tâches :**
 - Nombre de tâches en attente, assignées et en retard.
 - Performance individuelle des utilisateurs et des rôles (temps moyen de traitement des tâches).
- **Alertes et Notifications :**
 - Historique des alertes déclenchées (dépassement de seuils, erreurs d'API/BDD, tâches non assignées).
- **Rapports Personnalisés :**
 - Possibilité d'exporter les métriques et les données en formats CSV/PDF pour une analyse approfondie.
 - Génération de rapports sur les tendances historiques pour l'aide à la décision stratégique.

Ce tableau de bord servira d'outil central pour les administrateurs et les responsables, leur permettant de surveiller la santé d'APS, d'identifier les problèmes et de prendre des décisions éclairées pour maintenir l'efficacité opérationnelle.

Tableau des Rôles et Permissions

Ce tableau crucial résume les exigences révisées en matière de contrôle d'accès, illustrant la nouvelle granularité des attributions de processus. Il met en évidence la capacité du système à attribuer des processus à des groupes ou à des individus spécifiques, une flexibilité essentielle pour les opérations bancaires.

Rôle	Description	Permissions Clés
------	-------------	------------------

Adminis- trateur (Manager)	Responsable de la gestion complète du système de monitoring et de la résolution des problèmes APS.	Consultation des tableaux de bord, Configuration des alertes, Gestion des utilisateurs du monitoring, Déclenchement d'actions de maintenance (ex: archivage), Réaffectation manuelle des processus/tâches (à des groupes ou personnes).
Responsable de Groupe	Doit pouvoir consulter et agir uniquement sur les processus de son groupe attribué.	Consultation et action sur les processus de son groupe attribué, y compris la visibilité sur les processus attribués à des personnes spécifiques au sein de son groupe.

Tableau Récapitulatif des Solutions:

Ce tableau fournit un résumé concis de la manière dont chaque problème identifié est abordé par le système de monitoring, reflétant les dernières exigences et la réorientation stratégique. Il met en lumière le passage d'une visibilité générale à une détection proactive ciblée et la décision de maintenir la réaffectation des tâches manuelle.

Problème Adressé	Solution Proposée (Analyse d'Impact)	Fonctionnalité du Système de Monitoring Correspondante	Objectif Principal
Accumulation de données dans la base de données	Archivage automatique des processus obsolètes	Nettoyage des Processus Abandonnés	Réduire l'encombrement de la base de données et améliorer les performances.
Processus bloqué à la première tâche utilisateur	Réaffectation manuelle des tâches	Réaffectation Manuelle des Tâches	Assurer la continuité des processus et réduire les blocages liés aux utilisateurs par intervention manuelle.
Processus interrompu par une exception	Mise en place d'un système de reprise des processus	Mécanismes de Reprise	Augmenter la résilience du système face aux erreurs transitoires.
Lenteur générale des performances / Manque de visibilité en temps réel sur les processus problématiques	Détection et Gestion Proactive des Processus Problématiques	Surveillance et Détection Proactive des Processus Problématiques, Rapports et Tableaux de Bord	Offrir une visibilité ciblée et des outils d'analyse pour l'optimisation continue et l'intervention rapide.
Tâches non traitées / Utilisateurs inactifs	Système de notification automatique	Alertes et Notifications, Système de Notification Automatique	Rappeler les actions nécessaires et améliorer la réactivité des parties prenantes.