Annexe 1 - Préconisations à mettre en œuvre, réalisées dans le cadre d'un précédent audit

1. Sécurité du code

N°	Problématique	Plan d'actions	Criticité	Complexité
SEC1	Fonction de chiffrement des données GraphQL à modifier (AES-256-CBC vers AES-256-GCM).	Vérifier l'impact d'un changement d'algorithme de chiffrement. Si la décision est prise de modifier d'algorithme, effectuer le changement et appliquer les modifications nécessaires sur le code et/ou les données en conséquence de ce changement	Moyenne	Moyenne
SEC7	Regex coûteux avec possibilité de dénis de service	Simplifier les regex en question afin de réduire leur coût en ressource	Faible	Faible
SEC9	141 vulnérabilités de sécurité modérées et faibles	Traiter les vulnérabilités de sécurité détectées par Bearer	Moyenne	Moyenne
SEC10	Lancement automatique de SonarQube	Configurer la chaîne de CI pour lancer automatiquement une analyse de sécurité et de qualité de code	Faible	Faible

2. Qualité du code

N°	Problématique	Plan d'actions	Criticité	Complexité
QUA1	1 problème de fiabilité de code à priorité élevée	Analyser et traiter les problèmes de fiabilité	**Elevée**	Moyenne
QUA2	57 problèmes de fiabilité de code à priorité moyenne		Moyenne	Moyenne
QUA3	12 problèmes de fiabilité de code à priorité faible		Faible	Moyenne
QUA4	116 problèmes de maintenabilité à priorité élevée	Analyser et traiter les problèmes de maintenabilité	Moyenne	Faible
QUA5	346 problèmes de maintenabilité à priorité moyenne et 453 à priorité faible		Faible	Faible
QUA6	Conformité ESLint : 30 warnings bonnes pratiques de code	Analyser et traiter les 30 alertes de bonnes pratiques de code de niveau warning. Enregistrer ce commit spécifique de mise en conformité	Moyenne	Moyenne
QUA7	Configuration d'ESLint	Configurer ESLint pour traiter les 'warnings' en erreur. Optimiser la configuration par défaut d'ESLint. Intégrer les plugins propres à React et Typescript	Moyenne	Moyenne
QUA8	Lancement automatique d'ESLint	Installer la lib Husky Lancer une analyse ESLint à chaque commit	Faible	Faible
QUA9	Conformité formatage des fichiers source	Lancer prettier sur les 497 fichiers source non formatés suivant les règles. Enregistrer ce commit spécifique de mise en conformité	Faible	Faible
QUA10	Lancement automatique de Prettier	Installer la lib Husky Lancer un formatage automatique à chaque commit	Faible	Faible

3. Couverture des tests

N°	Problématique	Plan d'actions	Criticité	Complexité
COU1	Version de Cypress obsolète	Passer Cypress en v13 et mettre à jour les autres dépendances de la suite de tests	**Elevée**	Faible
COU2	Suite de tests désactivée dans la CI	Réintégrer l'exécution des tests dans la chaîne d'intégration continue	**Elevée**	Moyenne
COU3	Complexité excessive du projet de tests	Re-créer un sous-projet de tests avec une librairie plus moderne et plus performante. Simplifier la configuration et la mise en place des tests. Utiliser un environnement fixe et stable pour exécuter les scénarios de test	Moyenne	Moyenne

4. Documentation

N°	Problématique	Plan d'actions	Criticité	Complexité
DOC1	Revue du glossaire fonctionnel	Passer en revue la liste des termes et vérifier que les définitions sont à jour. Ajouter des termes importants qui seraient absents du glossaire	Moyenne	Faible
DOC5	Documentation de référence de l'API	Rédiger une documentation de référence sur les endpoints de l'API A minima sur les prochains endpoints développés. Mettre à disposition cette documentation sur une interface graphique	Moyenne	**Elevée**
DOC6	Procédure d'onboarding	Identifier les étapes de mise en place d'un environnement de travail. Rédiger une documentation guidant un nouvel intervenant dans son arrivée	Moyenne	Moyenne

5. Infrastructure (environnement, dépendances, CI/CD)

N°	Problématique	Plan d'actions	Criticité	Complexité
ENV1	Delta entre la base de données de prod et staging	Établir une procédure comportant la fréquence souhaitée de mise à jour des données de staging ainsi que la marche à suivre Concevoir un script effectuant l'opération pour réduire les coûts de maintenance	Faible	Moyenne
ENV3	Organisation confusante des buckets S3	Proposer une réorganisation des buckets S3 en respectant les contraintes imposées (notamment séparation des fichiers de facturation). Migrer les buckets actuels vers les nouveaux buckets réorganisés	Faible	Moyenne

ENV4	Bastion insuffisamment sécurisé	Solidifier la configuration SSH. Créer des utilisateurs spécifiques pour chaque intervenant. Configurer et activer UFW (firewall) Installer et configurer Fail2Ban	**Elevée**	Moyenne
ENV9	Accès au projet OpenStack	Vérifier les accès au projet OpenStack 'sfc-kingsman'. Désactiver les accès non légitimes	**Elevée**	Faible
DEP1	Incompatibilité des peer dependencies	Mettre à jour et ré-aligner les dépendances du projet. Ces deux tâches doivent être effectuées en même temps. Réinstaurer une installation classique de dépendances en retirant l'option --force-peer-deps.	**Elevée**	**Elevée**
DEP2	Dépendances obsolètes (47 majeures)	Étudier les changelogs des dépendances à faire évoluer. Réaliser les migrations de code nécessaires. Effectuer une recette complète de l'application pour détecter et corriger d'éventuelles régressions.	**Elevée**	**Elevée**
DEP3	Dépendances obsolètes (30 mineures)		Moyenne	Moyenne
DEP4	Dépendances suspicieuses (5 détectées)	Identifier des librairies pour remplacer les dépendances suspicieuses. Réaliser les migrations. Effectuer une recette complète de l'application pour détecter et corriger d'éventuelles régressions.	**Elevée**	**Elevée**
CHA1	Étape de test désactivée	Ré-activer l'étape de test dans la CI Vérifier son bon fonctionnement et corriger la suite de tests si besoin	**Elevée**	Moyenne
CHA2	Variable suspicieuse dans la configuration	Vérifier que la variable TOKEN_SECRET avec une valeur exemple n'est pas utilisée par quelconque service. Si elle est utilisée, changer sa valeur et passer la variable dans le store Gitlab	**Elevée**	Faible
CHA3	Vérifications TLS désactivées	Vérifier pourquoi la vérification TLS est désactivée pour Git et Docker. Si ces désactivations ne sont pas légitimes, remettre en place les vérifications TLS	**Elevée**	Faible
CHA4	Version de Docker-in-Docker fixée	Configurer un tag plus flexible sur cette image et vérifier son bon fonctionnement	Moyenne	Faible
CHA5	Optimisation et performances	Activer BuildKit et vérifier le bon fonctionnement de la chaîne Configurer le multi-stage dans la construction des images Docker. Paralléliser certaines étapes de la chaîne (notamment l'étape de test)	Faible	**Elevée**
CHA6	Configuration de la CI/CD insuffisamment documentée	Ajouter des commentaires expliquant les étapes du pipeline. Ajouter des commentaires justifiant pourquoi certaines étapes sont désactivées	Moyenne	Faible