

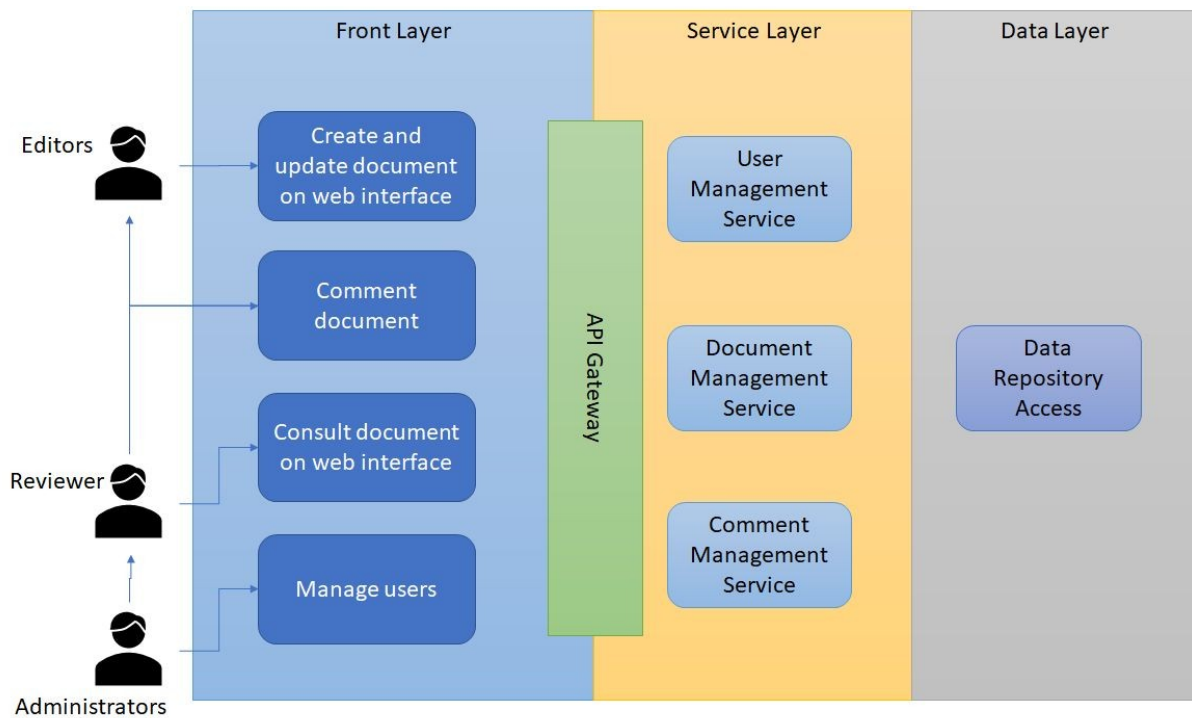


Titre du Projet				Réf. Du projet
	SCS GED			00_1SCSGED
Historique				
Version	Auteur	Description	Date	
1.0	Jules MESSOLO	Architecture Roadmap	30/10/2020	

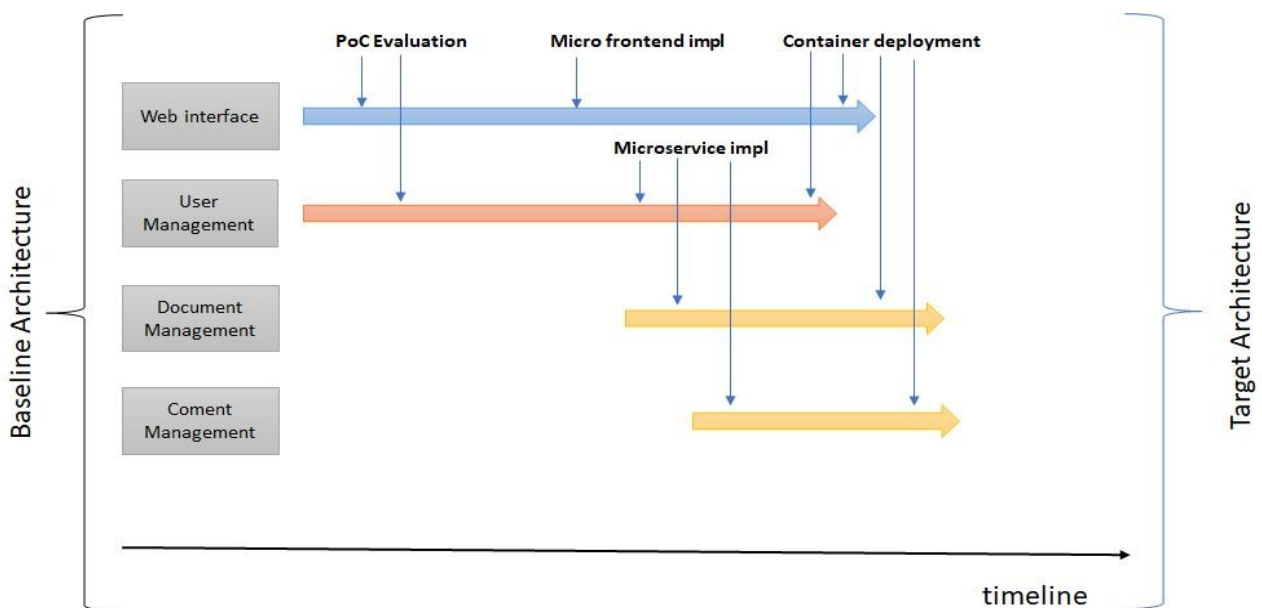
## Table des matières

Architecture Cible.....	3
Roadmap.....	3
1.Besoin de compétences.....	4
2.Répartitions des tâches et constitution des équipes.....	4
3.Privilégier l'agilité avec la méthode Scrum.....	4
4.Adopter une démarche DevOps.....	4
5.Mise en place un pipeline CI/CD.....	5
6.Opter pour une approche TDD.....	6
7.Implémentation de la couche front-end.....	7
8.Gestion de la Gateway.....	7
9.Microservice: Gestion des utilisateurs.....	7
11.Microservice: Gestion des commentaires.....	9
12.Mettre en place des outils de monitoring.....	10
13.Planning Prévisionnel.....	10
14.Le déploiement de la solution.....	11
15.Matrice des risques.....	11
16.Stratégie de mitigation.....	13

## Architecture Cible



## Roadmap



## 1. Besoin de compétences

La constitution de l'équipe de développement doit regrouper les compétences suivantes:

2 Développeur front-end et 2 Développeur Back-end

1 U.X Design

2 Ingénieur Système (ou Ingénieur DevOps)

1 Product Owner / Chef de projet

2 Business

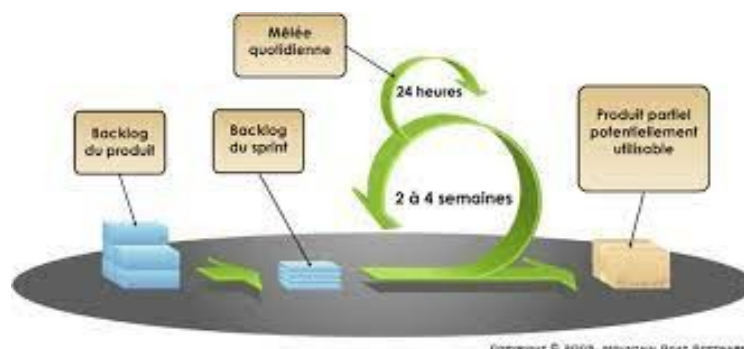
## 2. Répartitions des taches et constitution des équipes

Il est préconisé de permettre aux équipes constituée de développement d'être autonome et de pouvoir s'occuper de bout en bout de l'implémentation d'un bloc de construction de bout en bout. La solution étant décomposé en 5 grands bloc de construction.

## 3. Privilégier l'agilité avec la méthode Scrum

Pendant la phase d'implémentation de la solution il est préconiser d'utiliser une méthodologie agile, afin de régulièrement avec les équipes métiers faire un point sur les fonctionnalités implémenter, leur priorités et déceler éventuellement des difficultés dans la livraison et l'intégration des services.

Organiser le développement en itération, ponctué par des réunions et des démonstration pour valider le contenu des livrables. La solution sera incrémentée au fur et à mesure des sprints. Pour ce faire l'utilisation d'outil de gestion de projet comme Jira et confluence permettra aux équipes de mettre en place un **backlog évolutif** selon les priorités du clients ou les difficultés rencontrés. De découper son projet en feature, puis en user stories afin de bien couvrir toutes les fonctionnalités attendu.



## 4. Adopter une démarche DevOps

### Intégration continue

L'intégration continue est un ensemble de pratiques qui consiste à ce que les développeurs intègrent régulièrement leurs modifications de code à un référentiel centralisé afin de s'assurer que ces modifications ne produisent pas de régression dans l'application développée.

Des outils permettent de réunir les étapes nécessaires à la mise en place de l'intégration continue: **GitLab CI**.



### Livraison continue

La livraison continue est une méthode de développement de logiciels dans le cadre de laquelle les modifications de code sont automatiquement préparées en vue de leur publication dans un environnement de production à n'importe quel moment.

Elle permet aux développeurs d'automatiser les tests au-delà des simples tests d'unité, afin de vérifier différents aspects d'une mise à jour d'application. Il peut s'agir de tests d'interface, de charge, d'intégration, de fiabilité de l'API, etc. De cette manière, les développeurs peuvent vérifier de façon plus complète les mises à jour et détecter les éventuels problèmes à corriger avant le déploiement.

## 5. Mise en place un pipeline CI/CD

Il est préconisé d'avoir de mettre en place 4 environnements bien cloisonnés à savoir:

- Un environnement dédiés aux développeurs

- un environnement qui permet d'effectuer des test et de détecter éventuellement des régressions ou anomalies. Il est à disposition des testeurs et des équipes métiers. Il permet d'effectuer des démonstrations lors des livraisons de fin de sprint et par la suite des réajustements.
- Un environnement de pré-production (User Acceptance Test) permettant d'avoir un environnement qui se rapproche le plus de la production. Il permettra d'effectuer les test de charge, des test de performance, test utilisateurs.
- L'environnement de production.

De n'autoriser l'accès à chaque environnement qu'aux personnes concernés.

De planifier et communiquer les livraisons dans ces environnements.



## 6. Opter pour une approche TDD

Il est préconisé d'inclure des tests dans le déploiement de l'application et des opérations en cours. Une bonne stratégie de test aide à identifier les problèmes dans l'application avant même son déploiement

L'implémentation de la solution passe par l'écriture des différentes fonctionnalités qui débouche sur des scénarios puis des tests (TDD).

L'utilisation de framework comme Cucumber facilite la mise en place de cette approche.

SERVICE	Utilisateurs		
FONCTIONNALITÉ	Création de document	Réf	
Description	<b>En tant qu'utilisateur je peux créer un nouveau document</b>		
GIVEN (En tant que) utilisateur connecté WHEN (Quand) je clique sur le bouton création de document du menu THEN (alors) un nouveau document est créer.			

N° du Test	Fonctionnalité	Scénario	Action utilisateur	Action système	Résultat	Microservices
	Création de document	L'utilisateur connecté créer un document	L'utilisateur clique sur 'création documents'	Création d'un document.		Documents

## 7.Implémentation de la couche front-end

Architecturer la solution front-end

Il est important de penser enchaînement des écrans avant de commencer l'implémentation de la solution front-end.

S'assurer que l'interface utilisateur couvre toute les User stories présentent dans le backlog. Simuler les données afin de permettre aux développeurs front-end d'avancer en toute autonomie (JSON Server pour l'application Angular)

## 8.Gestion de la Gateway

La gestion de la passerelle intervient tôt dans l'implémentation de la solution.

L'API Gateway se doit d'être indépendante et se place entre le client et le Back-end et devient donc un point d'entrée à l'application.

Spring Cloud Gateway vise à fournir un moyen simple mais efficace d'acheminer vers les API et de leur fournir des préoccupations transversales telles que : la sécurité, la surveillance/les métriques et la résilience.

Les API Gateway ici seront séparées par domaine fonctionnel et par utilisation, voire par plateforme (web, mobile). L'implémentation de l'authentification est déléguée à la Gateway pour des question de charge et de maintenance. Attention la Gateway ne doit pas devenir un point de défaillance centralisé de l'application, pour cela elle doit être répliquée et « Load Balancée ».

L'implémentation de chaque microservice fait l'objet d'une 'feature' et celles ci constituent les jalons principales de la feuille de route du projet.

## 9.Microservice: Gestion des utilisateurs

Le microservice gestion des utilisateurs expose une API qui permet de récupérer, modifier des informations sur les utilisateurs possédant un compte. Cette API est privé et accessible uniquement par un client authentifié .C'est une API Restful utilise JSON comme format d'échange des informations. Il permet d'effectuer les taches suivantes.

Développement Back-	Développement	Taches
---------------------	---------------	--------

end	Front-end	
-----	-----------	--

Microservice	User Stories / Taches/ U.I	Point d'effort	Catégorie
Gestion des utilisateurs	En tant qu'utilisateur extérieur je peux me connecter sur la plateforme et accéder au service	12	input
	En tant qu'utilisateur extérieur l' U.I me permet de me connecter sur la plateforme et accéder au service	6	input
	Intégrer un système Authentification JWT		
	Intégrer un système d'autorisation (Rôles)		
	En tant qu'utilisateur extérieur je peux demander une réinitialisation de mon mot de passe.	12	input
	En tant qu'utilisateur extérieur l' U.I me permet de demander une réinitialisation de mon mot de passe.	6	input
	En tant qu'admin je peux créer un compte pour un nouveau collaborateur.	6	admin
	En tant qu'admin l' U.I me permet de créer un compte pour un nouveau collaborateur.	6	admin
	En tant qu'admin je peux supprimer le compte d'un collaborateur.	9	admin
	En tant qu'admin l' U.I me permet de supprimer le compte d'un collaborateur.	9	admin

## 10. Microservice: Gestion des documents

Le microservice gestion des documents expose une API qui permet gérer les documents associés à un compte utilisateur. Cette API est privé et accessible uniquement par un client authentifié .C'est une API Restful utilise JSON comme format d'échange des informations. Il permet d'effectuer les taches suivantes.

Microservice	User Stories / Taches/ U.I	Point d'effort	Catégorie
Gestion des documents	En tant qu'utilisateur je peux créer un nouveau document et y accéder	6	input
	En tant qu'utilisateur l' U.I me permet de créer un nouveau document et y accéder	6	
	En tant qu'utilisateur je peux modifier un document.	9	input
	En tant qu'utilisateur l' U.I me permet de modifier un document.	6	
	En tant qu'utilisateur je peux indiquer que mon document doit être validé (demande de validation)	9	input
	En tant qu'utilisateur l' U.I me permet d' indiquer que mon document doit être validé (demande de validation)	6	
	En tant qu'utilisateur je peux être notifié (par mail) à	9	input



	chaque modification d'un de mes documents "en cours de revue".		
	En tant qu'utilisateur l' U.I me permet de demander à être notifié (par mail) à chaque modification d'un de mes documents "en cours de revue".	6	input
	En tant qu'utilisateur extérieur je peux visualiser ma liste de documents	6	input
	En tant qu'utilisateur extérieur l' U.I me permet de visualiser ma liste de documents	9	input
	En tant qu'utilisateur extérieur je peux trier par état mes documents.	6	
	En tant qu'utilisateur extérieur l' U.I me permet de trier par état mes documents.	6	input
	En tant qu'utilisateur je peux être notifié (par mail) à chaque changement d'état du document.	12	input
	En tant qu'utilisateur avec un rôle 'validation' je peux visualiser la liste des documents à évaluer (état : en cours de revue).	12	input
	En tant qu'utilisateur avec un rôle 'validation' Je peux indiquer que le document est prêt pour publication (état clôturé).	9	input
	En tant qu'utilisateur avec un rôle 'validation' Je peux télécharger un document au format PDF.	12	admin
	En tant qu'admin Je peux donner l'accès d'un ou plusieurs documents à un utilisateur	9	admin

Microservices	User Stories / Taches/ U.I	Point d'effort	Catégorie
Gestion des versions	En tant qu'utilisateur extérieur Je peux indiquer la version de mon document (versionner)	12	input
	En tant qu'utilisateur extérieur Je peux visualiser une version antérieure de mon document.	6	review
	En tant qu'utilisateur avec un rôle 'validation' Je peux indiquer une version sur le document et visualiser les versions antérieures.		review

## 11. Microservice: Gestion des commentaires

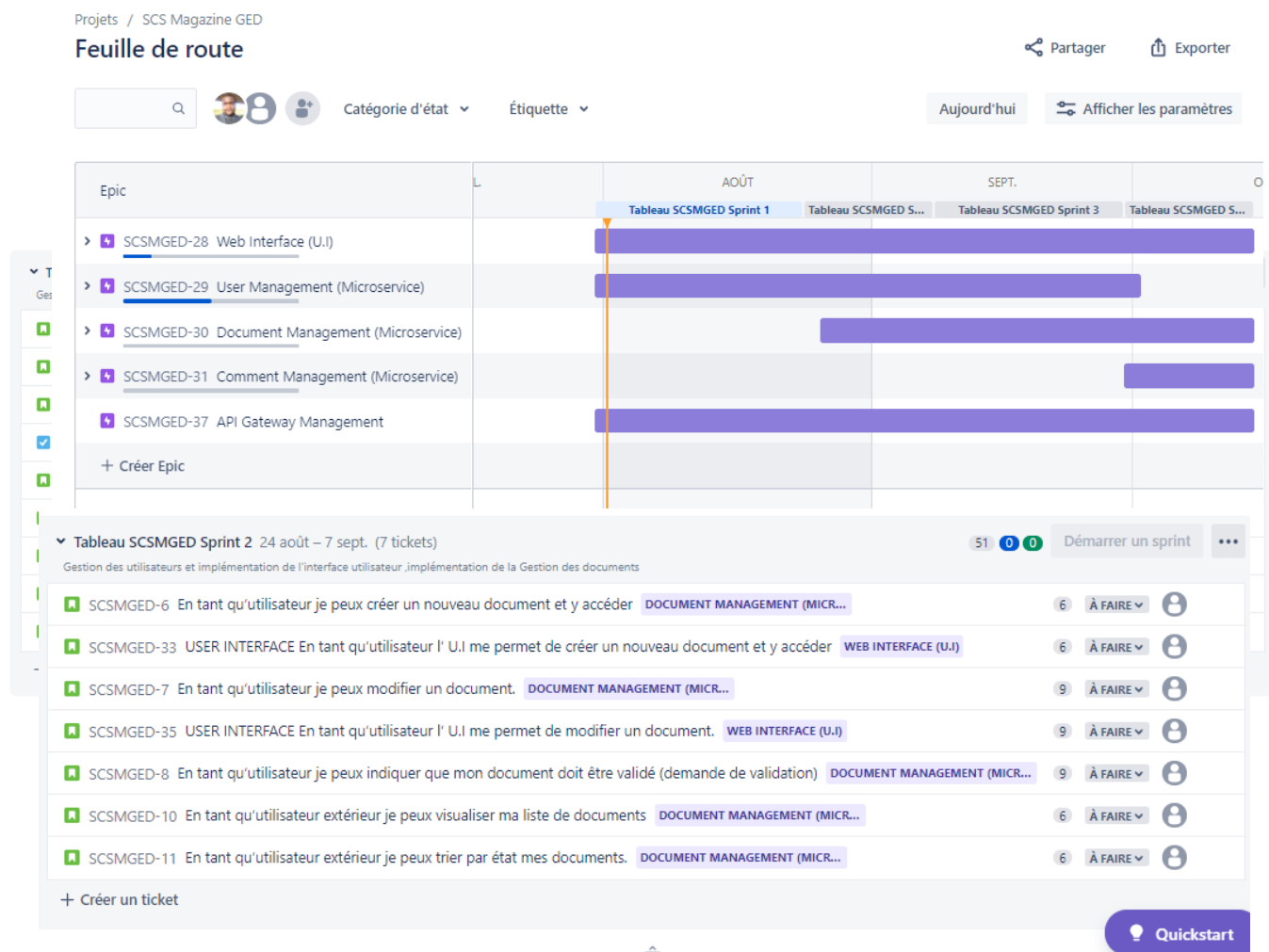
Microservices	Tache	Point d'effort	Catégorie
Gestion des commentaires	En tant qu'utilisateur je peux créer un commentaire et le diffuser	6	review
	En tant qu'utilisateur je peux répondre à un commentaire.	6	review
	En tant qu'utilisateur avec un rôle 'validation' Je peux ajouter et répondre à des commentaires.		review

## 12.Mettre en place des outils de monitoring

Pour surveiller le bon fonctionnement de chaque microservices il est important de suivre son fonctionnement à l'aide d'outil de monitoring. Nagios

## 13.Planning Prévisionnel

L'implémentation de la solution est découpée en 4 sprints de 2 à 3 semaines. Chaque sprint est ponctué par une démonstration, une livraison en environnement de Test et des ateliers de formation utilisateurs.



## 14. Le déploiement de la solution

Le déploiement de la solution fait l'objet d'une 'feature' dans la feuille de route du projet . Il reprend le déploiement de chaque microservice dans son conteneur qui peuvent être intégré au backlog comme des user stories ou des tâches.

Celui n'intervient qu'à la fin de l'implémentation complet du microservice en question.

## 15. Matrice des risques

**Probabilité (P):** une probabilité, c'est évaluer la possibilité que le risque devienne un problème.

1. inexistante 2. Peut se produire 3. Forte chance de se produire 4. Très forte chance de se produire 5. Inéluctable

**Gravité (G):** La gravité exprime l'impact du risque devenu problème sur le déroulement du projet.

1. Anodin 2. Pas préjudiciable 3. Peu préjudiciable 4. Préjudiciable 5. Extrêmement préjudiciable

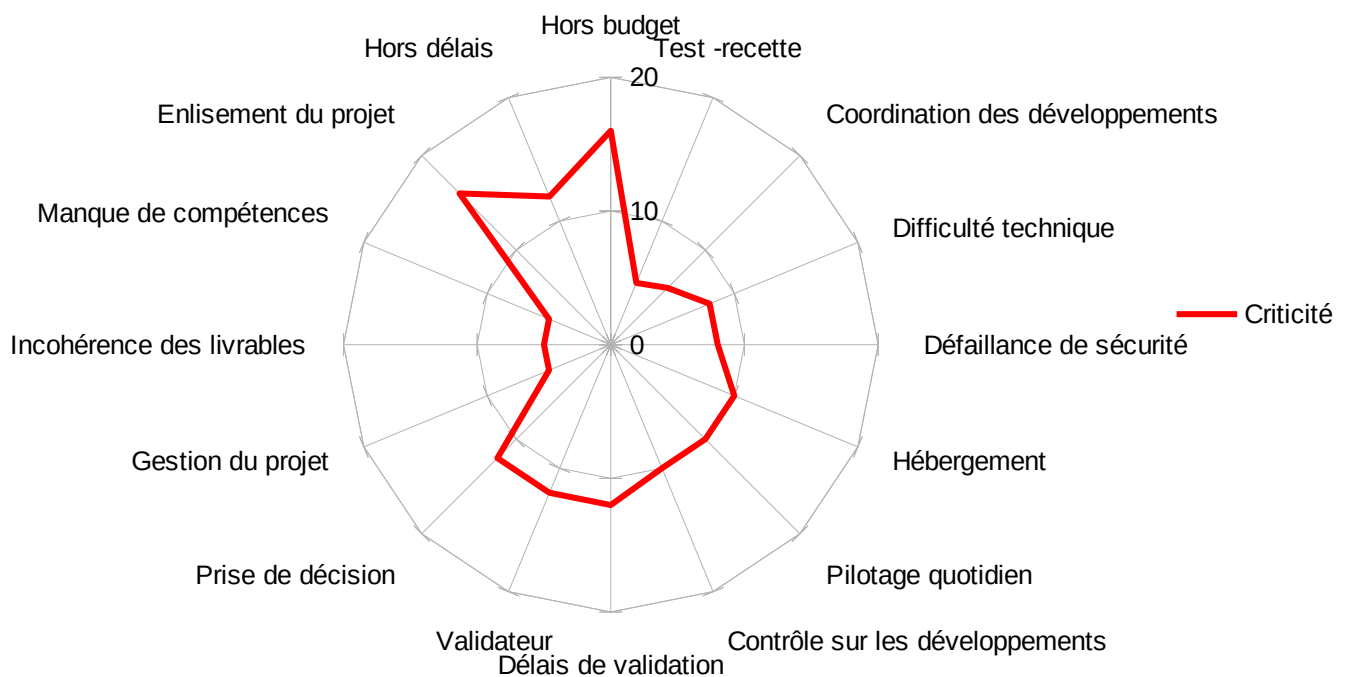
**Criticité (C) = Probabilité (P) X Gravité (G)**

1 à 5	6 à 10	11 à 15	15 <
-------	--------	---------	------

Catégorie	Risque identifiés	Titre	P	G	C	Responsabilité
BUDGET	Risque de dépassement de budget	Dépassement budget	4	4	16	Product Owner / Chef du projet
DÉLAIS	Risque de dépassement des délais.	Dépassement délais	3	4	12	Product Owner / Chef de projet
DÉLAIS	Risque d'enlèvement du projet	Enlèvement du projet	4	4	16	Product Owner
ÉQUIPE	Absence de compétences nécessaires aux bon déroulement du projet.	Compétences manquantes	1	5	5	SuperLoftTech
PÉRIMÈTRE	Risque de livrable inadaptée ou incohérent	Incohérence des livrables	1	5	5	Product Owner /SuperLoftTech

	avec les attentes.					
PÉRIMÈTRE	Mauvaise gestion du projet.	Gestion du projet	1	5	5	SuperLoftTech
PÉRIMÈTRE	Absence de prise de décision.	Prise de décision	3	4	12	SCS Magasin
DÉCISION	Mauvaise identification d'un validateur.	Valideur	4	3	12	SCS Magasin
DÉLAIS	Allongement des délais de validation.	Délais de validation	3	4	12	Chef de projet/ Product Owner
QUALITÉ	Défaut de contrôle sur le développement.	Contrôle sur les développements	2	5	10	Chef de projet
ÉQUIPE	Freins dans le pilotage opérationnel quotidien	Pilotage quotidien	2	5	10	Chef de projet/ Product Owner
QUALITÉ	Réalisation non compatible avec l'environnement cible.	Hébergement	2	5	10	Chef de projet
SÉCURITÉ	Défaillance en terme de sécurité	Défaillance de sécurité	2	4	8	SCS Magasin
QUALITÉ	Absence de références techniques en cas de difficulté	Difficulté technique	2	4	8	Chef de projet
ÉQUIPE	Manque de coordination dans le développement mobile et web	Coordination des développements	3	2	6	Chef de projet/ Product Owner
QUALITÉ	Réalisation d'un produit non fonctionnel	Test -recette	1	5	5	Chef de projet - Product Owner

## Gestion des Risques



## 16.Stratégie de mitigation

Titre	Origine du risque	Prévention	Réparation	Mitigation
Dépassement budget	Absence de budget établi	Prioriser les fonctionnalités à implémenter - Limiter les ressources disponibles sur le projet à 105 j/h (+/- 10%)		Réduction
Dépassement délais	Absence de calendrier prévisionnelle d'implémentation. Estimation des charges de travail à réaliser.	Absence de calendrier prévisionnelle d'implémentation. Estimation des charges de travail à réaliser.	Établir un calendrier prévisionnel. Estimation des charges.	Réduction
Enlisement du projet	Pas de Planning même prévisionnel	Pas de Planning même prévisionnel		Réduction

<b>Compétences manquantes</b>	Toutes les forces de production n'ont pas encore affecté au projet	Toutes les forces de production n'ont pas encore affecté au projet	Estimer les charges et les compétences nécessaires pour le projet	Réduction
<b>Incohérence des livrables</b>	Absence d'exigence sur le contenu des livrables. Aucune Date de livraison.	Absence d'exigence sur le contenu des livrables. Aucune Date de livraison.	Établir une feuille de route quand au contenu des livrables	Réduction
<b>Gestion du projet</b>	Absence de méthodologie, Organisation non établi.	Absence de méthodologie, Organisation non établi.	Demander au prestataire de communiquer sur sa méthodologie le déroulement de la phase d'implémentation	Réduction
<b>Prise de décision</b>	Les décisionnaires ne sont pas identifiés ni les rôles formellement établis. Pas de P G I	Les décisionnaires ne sont pas identifiés ni les rôles formellement établis. Pas de P G I	Établir un Plan de gestion des intervenants. Identifier clairement les responsabilités	Réduction
<b>Valideur</b>	Pas de Plan Gestion des Intervenants établi. Aucune assignation des responsabilités	Pas de Plan Gestion des Intervenants établi. Aucune assignation des responsabilités	Assigner clairement les responsabilités et les communiquer de façon officiel	Réduction
<b>Délais de validation</b>	Responsabilités non attribués et calendrier non établi	Responsabilités non attribués et calendrier non établi	Assigner clairement les responsabilités et les communiquer de façon officiel via un P.G.I	Réduction
<b>Contrôle sur les développements</b>	Absence de suivi des développements par le chef de projet. Défaillance sur le contrôle des réalisations	Absence de suivi des développements par le chef de projet. Défaillance sur le contrôle des réalisations	S'assurer de la qualité du chef de projet	Réduction
<b>Pilotage quotidien</b>	La méthodologie d'implémentation n'a pas encore défini.	La méthodologie d'implémentation n'a pas encore défini.	Communiquer sur la méthodologie. Choisir une méthodologie d'implémentation	Réduction
<b>Hébergement</b>	Information sur le S.I d'hébergement manquante ou non communiqué.	Information sur le S.I d'hébergement manquante ou non communiqué.	Communiquer sur l'architecture du S.I d'hébergement. Représenter les différentes interactions avec les éléments extérieurs	Réduction
<b>Défaillance de sécurité</b>	Norme de sécurité non respecté. Spécifications techniques mal implémenté	Norme de sécurité non respecté. Spécifications techniques mal implémenté	S'assurer de la bonne compréhension des exigences fonctionnelles et technique en terme de sécurité. Réunion pour mise au point.	Réduction
<b>Difficulté technique</b>	Manque d'expérience des équipes de développement	Manque d'expérience des équipes de développement	S'assurer de l'expérience du personnel affecté sur projet. Mettre en place des sessions de formation	Réduction
<b>Coordination des développements</b>	Travaille de coordination et de prise d'information absent. Pas de remonté d'information.	Travaille de coordination et de prise d'information absent. Pas de remonté d'information.	Répétition de réunion sur l'avancement des taches. Partage des repository .	Réduction

Test -recette	Plan de Test Absent. Stratégie de test à établir	Plan de Test Absent. Stratégie de test à établir	Écrire une stratégie de test et plan de test à établir	Réduction
---------------	--	--	---	-----------

