

# Plateforme Agora

18/04/2024

## FICHE DE SUIVI DES RISQUES

Version: 2
Date: 30/05/2024
Rédigé par: Yacine Ben Ahmed
Relu par: Yacine Ben Ahmed, Corentin Pille, Mohamed Cherfi, Anass El Gharbaoui, Sid Ahmed Brahimi, Robin Sapin, Michel Nassalang
Approuvé par:
Signature:

1. Réévaluation et suivi des risques initiaux	2
2. Nouveaux risques identifiés	4
3. Rapports d’incident	5

## 1. Réévaluation et suivi des risques initiaux

Ref.	Catégorie	Description	Gravité (Initial -> Actuelle)	Fréquence sur les 6 premiers sprints	Les actions préventives/curatives ont-elles été appliquées et/ou modifiées?
R1.1	Client	Le client peut ne pas être disponible pour un rendez-vous, ce qui peut retarder le début de la correction du livrable	2 → 2	0%	Préventive: Appliqués.  Curative: Aucun problème sur la présence du client.
R2.1	Contrat	Une charge de développement importante peut entraîner un retard de livrables	3 → 2	33.3%	Préventive: Appliqués.  ----- Curative: Appliqués + on s'est permis à deux reprises d'élargir nos sprints à 3 semaines au lieu de 2.
R2.2	Contrat	Certains aspects des fonctionnalités peuvent ne pas avoir été compris ce qui peut influencer sur leur réalisation.	3 → 3	16.6%	Préventive: Appliqués  ----- Curative: Appliqués.
R3.1	Organisation	Les membres du projet n'ont pas les mêmes disponibilités pour les réunions	2 → 1	100%	Préventive: Appliqués.  ----- Curative: Appliqués + on propose la visioconférence via Slack pour les membres absents.

R3.2	Organisation	Une répartition non équitable et qui met en difficulté certains membres du projet	3 → 2	50%	<p>Préventive: Partiellement appliqués, on ne fait pas de planning poker mais on estime quand même les coûts des tâches pour les répartir équitablement.</p> <p>-----</p> <p>Curative: Appliqués.</p>
R3.3	Organisation	Le travail est collectif et tout défaut de communication impactera la qualité du résultat final.	3 → 3	16.6%	<p>Préventive: Appliqués.</p> <p>-----</p> <p>Curative: Appliqués.</p>
R4.1	Technique	Les technologies utilisées peuvent ne pas bien être maîtrisées.	3 → 2	100%	<p>Préventive: Appliqués à l'aide d'un tutoriel Youtube de 10h complet sur Symfony suivi par l'ensemble du groupe.</p> <p>-----</p> <p>Curative: Partiellement appliqués, nous n'avons pas encore sollicité le référent technique dans la phase de développement.</p>
R4.2	Technique	La plateforme présente des bugs rendant certaines fonctionnalités inutilisables.	2 → 2	100%	<p>Préventive: Appliqués.</p> <p>-----</p> <p>Curative: Appliqués. En l'état actuel, pas de bug définitif sur la plateforme..</p>
R4.3	Technique	Les deux équipes présentent des problèmes pour s'accorder sur l'intégration des jeux sur la plateforme.	4 -> 4	50%	<p>Préventive: Appliqués partiellement, la "boîte noire" était encore très floue pendant l'étude du projet.</p> <p>-----</p> <p>Curative: Appliqués.</p>

Echelle de gravité : 1 = négligeable, 2 = non négligeable, 3 = forte, 4 = très forte

## 2. Nouveaux risques identifiés

Fiche de suivi du risque R3.4	
Description	Certains membres de l'équipe ne participent pas oralement aux réunions avec le client et autres intervenants externes donnant une mauvaise image du groupe.
Criticité	2
Fréquence	100%
Date d'établissement	18/04/2024
Solution préventive	Préparer en amont les réunions avec un étalement de la parole.
Solution curative	Lors des réunions client, chaque personne est amenée à présenter lui-même ce qu'il a développé.

Fiche de suivi du risque R3.5	
Description	Désaccord décisionnelle avec l'autre équipe sur une partie commune du projet
Criticité	3
Fréquence	33.33%
Date d'établissement	18/04/2024
Solution préventive	Organiser une réunion où les 2 partis exposent leur arguments afin de clarifier les points de vues et décider
Solution curative	Laisser le client ou le référent technique décider (selon le type de désaccord)

Fiche de suivi du risque R4.4	
Description	Interruption de GitLab
Criticité	1
Fréquence	50%
Date d'établissement	18/04/2024
Solution préventive	Ignorer: On pourra toujours push sur GitLab lorsque l'interruption sera finie (généralement ~30 minutes)
Solution curative	Ignorer.

### 3. Rapports d'incident

Identifiant: RI1
Période de l'incident: 29/01/2024 - 11/02/2021 (1er sprint)
Description: Répartition des tâches trop désorganisée sur ce sprint. Trop de gens ont des mini-tâches qui dépendent d'autres tâches et personne ne peut avancer efficacement. (chaque tâche doit être retouchée)
Conséquences: Le sprint n'a pas été très rentable, la majorité des tâches n'ont pas été effectuées entièrement.
Recommandation: Rassembler les tâches dépendantes dans une même tâche (avec des sous-tâches du coup) et organiser le groupe en équipes pour les réaliser.

Identifiant: RI2
Date de l'incident: 13/03/2024
Description: Le groupe Jeux a rassemblé pleins de dossiers dans un nouveau répertoire "app/" sans nous consulter.
Conséquences: Impossibilité pour nous de merge sur preprod car pleins de conflits étaient provoqués à cause des namespaces qui ne correspondaient plus.
Recommandation: Toujours s'accorder entre différentes d'équipes d'un même projet en cas de changement d'architecture concernant impliquant différentes équipes d'un projet.

Identifiant: RI3
Date de l'incident: 28/03/2024
Description: Les deux groupes n'étaient pas d'accord concernant la doc: le groupe Jeux souhaitait deux docs bien distinctes alors que le groupe Plateforme souhaitait une doc commune et uniforme.
Conséquences: Notre groupe Plateforme a attendu quelques jours une réponse du client par rapport à ce souci. (au final: deux docs distinctes mais une rédaction uniforme)
Recommandation: Clarifier ce point le plus tôt possible avec le client lors de l'étude du projet.

Identifiant: RI4
Date de l'incident: 16/04/2024
Description: Système d'invitation non conforme à ce que le client souhaitait. Il souhaitait une expiration en une semaine ainsi qu'une notification au joueur lors de l'expiration de l'invitation, ce qui n'était pas le cas.
Conséquences: Refonte nécessaire et déception du client (qui remet en question notre gestion de projet)
Recommandation: Il faut toujours demander le maximum de détails au client concernant une fonctionnalité avant de la développer, il ne doit pas y avoir d'ambiguïté.

Identifiant: RI5
Date de l'incident: 07/05/2024
Description: Réunion client présentant une multitude de bugs (exclusion de joueur buggué, possibilité de grader quelqu'un en Admin dans le front, bannissement non implémenté mais présent, bug lors de l'acceptation d'une invitation, noms des comptes générés non conforme à ce qu'aurait voulu le client)
Conséquences: Client insatisfait du rendu du produit + planification d'un sprint d'urgence pour corriger les bugs ⇒ extension de la date prévue de fin de projet
Recommandation: Toujours tester un maximum ses fonctionnalités avant de les présenter au client. Si une fonctionnalité présente un bug, elle ne doit pas être présentée. Aussi, toujours demander des précisions et ne pas prendre d'initiatives de soi-même afin d'être conforme aux volontés du client.