

Liste des risques

2. Risques

2.1. L'équipe n'arrive pas à apprendre les technologies suffisamment rapidement

2.1.1. Risk ranking

élevé

2.1.2. Description

L'équipe est novice, ne connaît pas les technologies et n'arrive pas à les maîtriser suffisamment rapidement.

2.1.3. Impacts

Tout ou une partie du projet ne fonctionne pas

2.1.4. Indicateurs

Retour négatif lors des réunions hebdomadaires ou le projet n'avance plus.

2.1.5. Mitigation strategy

L'accent a été mis sur l'apprentissage des technologies, notamment au début du projet.

2.1.6. Contingency plan

On peut simplifier la technologie si possible ou revoir les objectifs à la baisse.
On mobilise toute l'équipe sur la technologie problématique.

2.2. Un membre de l'équipe est soudainement indisponible

2.2.1. Risk ranking

élevé

2.2.2. Description

Un membre de l'équipe dont l'aide est indispensable est indisponible.

2.2.3. Impact

Cela bloque l'avancée du projet, les autres membres ne pouvant plus travailler.

2.2.4. Indicateurs

2.2.5. Mitigation strategy

On s'est renseignée à l'avance la disponibilité des membres, on essaye, si possible, d'avoir 2 personnes informées sur les mêmes technologies.

2.2.6. Contingency plan

Le membre le plus adapté essaye de comprendre la technologie impliquée.
Essaye d'avancer les choses indépendantes.

2.3. Deux membres de l'équipe ne sont pas d'accord sur un sujet

2.3.1. Risk ranking

faible

2.3.2. Description

Deux membres ont une opinion divergente sur une partie de l'implémentation

2.3.3. Impact

Le projet prend du retard, le temps de trouver un consensus.

2.3.4. Indicateurs

2.3.5. Mitigation strategy

On a défini la plupart des caractéristiques au préalable en équipe.

2.3.6. Contingency plan

L'équipe vote, la majorité gagne. Si égalité il y a, le project leader décide.

2.4. On n'arrive pas lier le backend et le frontend

2.4.1. Risk ranking

élevé

2.4.2. Description

L'interface graphique ne fonctionne pas avec le backend

2.4.3. Impact

Le projet n'aboutira pas à une application déployable et livrable.

2.4.4. Indicateurs

Malgré plusieurs tentatives et beaucoup de temps, on ne trouve pas de solutions.

2.4.5. Mitigation strategy

Anticiper la difficulté et commencer tôt à investiguer la technologie.

2.4.6. Contingency plan

On mobilise toute l'équipe pour faire fonctionner le projet quitte à sacrifier d'autres features (les moins importantes)

2.5. On n'arrive pas à finir dans les délais impartis

2.5.1. Risk ranking

élevé

2.5.2. Description

Malgré le travail de l'équipe, les features basiques requises ne sont pas terminées.

2.5.3. Impact

L'objectif final n'est pas atteint et l'application ne fonctionne potentiellement pas.

2.5.4. Indicateurs

L'équipe prend beaucoup de retard sur le plan initial.

2.5.5. Mitigation strategy

Commencer par les parties les plus importantes du projet.

2.5.6. Contingency plan

On rend le projet dans l'état non terminé.

2.6. Problèmes légaux liés aux données utilisateur

2.6.1. Risk ranking

faible

2.6.2. Description

On ne peut pas exploiter les données des utilisateurs.

2.6.3. Impact

On crée moins de profit pour l'application.

2.6.4. Indicateurs

On remarque que les utilisateurs n'acceptent pas de donner leurs données.

2.6.5. Mitigation strategy

On charge une personne de contrôler cet aspect.

2.6.6. Contingency plan

On cherche d'autres moyens de gagner de l'argent (via les pubs)

2.7. On n'arrive pas à mettre des publicités

2.7.1. Risk ranking

faible

2.7.2. Description

L'application n'a pas assez d'influence et on ne trouve pas d'annonceurs.

2.7.3. Impact

On crée moins de profit pour l'application.

2.7.4. Indicateurs

L'application n'est pas suffisamment visitée et cela décourage les annonceurs

2.7.5. Mitigation strategy

On fait de la pub pour notre application afin qu'elle soit plus visitée

2.7.6. Contingency plan

On cherche d'autres moyens de gagner de l'argent.

2.8. On ne connaît pas l'architecture microservices

2.8.1. Risk ranking

modéré

2.8.2. Description

L'équipe ne connaît pas l'architecture microservices.

2.8.3. Impact

L'équipe n'arrive pas à mettre en place cette architecture.

2.8.4. Indicateurs

Le projet n'avance pas à cause d'une mauvaise architecture

2.8.5. Mitigation strategy

On se renseigne sur les microservices et on s'inspire du projet d'exemple

2.8.6. Contingency plan

On demande de l'aide.