**Liste des Risques**

**Développement collaboratif**

**Acteurs :**

Nicolas Buffon : jokahero@gmail.com

Marc Plano-Lesay : marc.planolesay@gmail.com

Jean-Christophe Ricard : jch.ricard@gmail.com

Adrien Vetillart : adrien.vetillart@gmail.com

**Versions :**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Auteur | Date | Description |
| 1.0 | Adrien Vetillart | 24/11/2011 | Etablir la liste des principaux risques |

**Introduction**

Ce document permet de lister l’ensemble des risques liés à notre projet et pouvant perturber son bon déroulement. Ces risques peuvent être de différents types : humains, technologiques, liés aux outils etc. Dans ce document, les risques seront caractérisés par leur type, une description, un niveau de difficulté et une solution à envisager afin de les surmonter le plus efficacement possible. Voici les quatre niveaux de risques que nous avons choisis : Facile, Intermédiaire, Difficile et Bloquant. A noter qu'aucun risque bloquant n'a été relevé. Quoiqu'il en soit, le plus important reste de mettre en œuvre des solutions appropriées.

I. Risques liés aux outils utilisés 5

A. RSA (Rational Software Architect) 5

B. Git 5

II. Risques liés aux technologies utilisées 6

A. Génération de code avec RSA 6

III. Risques liés à l'organisation du projet 7

A. Planning serré avec une marge d'erreur faible 7

III. Conclusion 8

# I. Risques liés aux outils utilisés

Les risques suivants sont liés en grande partie à l’utilisation d’outils non connus par tous les acteurs du projet ainsi qu'à un manque de connaissance. Ils ont donc pour conséquence une perte de temps et ils n’ont pas d’impacts directs sur le fonctionnement de l’application (liés au développement).

## RSA (Rational Software Architect)

**Niveau de risque** : Intermédiaire

**Description** : Afin de concevoir et développer cette application, nous allons utiliser un nouvel outil que nous ne maitrisons pas dans sa globalité. En effet, aucun des membres de l’équipe ne possède d’expérience antérieure avec RSA que ce soit à titre personnel ou professionnel. Ce logiciel étant relativement complexe et complet, il nécessite un certain temps d’adaptation afin de pouvoir le prendre en main.

**Conséquences**: Une perte de temps conséquente si l’on fait la somme du temps perdu par l’ensemble des membres de l’équipe.

**Stratégie de traitement du risque** : La solution à ce risque reste la « communication » entre les différents membres. En effet, un acteur du projet qui rencontre un problème majeur sur le logiciel doit demander de l’aide aux autres collaborateurs s’il se retrouve bloqué et il se doit d’établir une trace de la solution (dans un document annexe, interne à l’équipe) si possible. Ce document servira ainsi de référence pour l'équipe en ce qui concerne l'utilisation des différents outils.

## Git

**Niveau de risque** : Facile

**Description** : Le serveur SVN que nous avons choisis pour développer notre projet est Git. Deux des quatre membres de l’équipe ne connaissent pas cet outil mais possède des bases avec d’autres systèmes de gestion de fichiers. Le temps d’adaptation devrait être relativement rapide.

**Conséquences**: Légère perte de temps pour les deux membres concernés.

**Stratégie de traitement du risque** : Entraide entre les différents acteurs de l’équipe (transmission de connaissances) ainsi qu’un travail personnel de la part des membres concernés (Adrien Vetillart et Jean Christophe Ricard) pour apprendre les notions propres à l’outil.

# II. Risques liés aux technologies utilisées

Les risques de cette partie sont ceux liés à l’utilisation des technologies et qui peuvent avoir un impact direct sur le développement de l’outil.

## A. Génération de code avec RSA

**Niveau de risque** : Intermédiaire

**Description** : Ce risque est difficilement évaluable étant donné que pour le moment la conception et le développement n’ont pas commencé sous RSA. Cependant, en nous reposant sur nos expériences précédentes et sur la description et le fonctionnement du logiciel, on ne peut omettre le fait que les risques liés à la génération de code à partir de diagrammes est omniprésent. En effet, utilisé un outil comme RSA demande une syntaxe et des éléments rigoureux et précis qui ne sont pas forcément maitrisés à cent pour cent par l’ensemble des acteurs. Il existe toujours des risques de bugs ou crash lors de la génération de code automatique.

**Conséquences** : Perte de temps et de données lors de bugs ou crashs.

**Stratégie de traitement du risque** : Penser à sauvegarder le plus souvent possible le travail réalisé pour éviter les pertes de données en cas de problèmes. En ce qui concerne le logiciel lui-même, prendre connaissance des fonctionnalités qui ne fonctionnement pas correctement (se référer à la documentation officielle ou au fournisseur) et essayer de mettre en place une solution de rechange pour faire la même action d’une autre manière. L’étape d’observation sera réalisée par Nicolas Buffon (qui est la première personne a travaillé sous RSA pour réaliser le diagramme des UC), qui se chargera de former l’équipe par la suite.

# III. Risques liés à l'organisation du projet

Nous allons voir quels sont les risques que nous avons relevé, liés à l'organisation du projet.

## A. Planning serré avec une marge d'erreur faible

**Niveau de difficulté** : Difficile

**Description** : Ce risque est lié aux exigences client et à la solution que nous avons proposée. En effet, la demande étant précise sur les principaux UC à traiter mais assez large pour les prises d'initiatives (pour les options annexes, optionnelles ainsi que les moyens pour développer l'application), nous avons élaborer une solution qui ne laisse pas beaucoup de place pour les erreurs d'organisation et la perte de temps. En effet, nous sommes soucieux d'un travail et d'un produit fini bien fait et nous avons donc décidé de serrer le planning.

**Conséquences** : Risque de retard sur des documents à rendre ou sur certaines phases. Léger risque de livrer une application avec des fonctionnalités non codées par rapport à ce qui a été présenté au client (dans les divers documents) lors des premières phases.

**Stratégie de traitement du risque** : Il va falloir respecter les plannings et les tâches prévus dans le temps fixé au préalable et jugé adéquate. Des réunions d'équipe supplémentaires pourront être mises en place afin de discuter des problèmes rencontrés et ainsi pouvoir rebondir le plus rapidement possible sur des solutions viables. Le chef d'équipe jouera donc un rôle clés dans la communication entre les différents acteurs et s'assurera que chaque personne respecte les délais et les conditions établis.

# III. Conclusion

Nous avons vu au travers de ce document, que la réalisation du projet RSA n'était pas sans risques. Il faut donc que chaque membre de l'équipe soit conscient de cette situation. Chacun sera confronté à des difficultés lors des différentes phases de ce projet et il faudra avoir à l'esprit que tout écart (de temps, manque de ressources etc.) devra être relevé au plus vite afin de mettre en place une solution adaptée après concertation des quatre collaborateurs. Cependant, il est difficile de faire une liste exhaustive de tous les risques liés au projet car il existe toujours une part variable liée à l'imprévu. Pour finir on peut dire que la communication, la motivation, l'entraide et l'organisation seront les maitres mots de notre équipe afin de pouvoir surmonter ces problèmes.