

**Équipe No. 3**

**Projet Intégrateur 3**  
**Plan de projet**

**Version 1.2**

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

## Historique des révisions

Date	Version	Description	Auteurs
2017-01-18	1.0	Complétion des points 1 (introduction), 2 (énoncé des travaux) et 5 (équipe de développement).	Desjardins, Simon-Pierre Grégoire, Frédéric Villeneuve, Ulric
2017-01-30	1.1	Rédaction de la gestion de configuration, l'entente contractuelle, la gestion de risque, la gestion des exigences, le contrôle de la qualité et l'échéancier du projet	Villeneuve, Ulric Desjardins, Simon-Pierre Grégoire, Frédéric
2017-02-01	1.2	Changement des heures-personne pour la section échéancier, ajustement de la gestion des risques et mise en page.	Villeneuve, Ulric Desjardins, Simon-Pierre Grégoire, Frédéric

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

## Table des matières

1.	Introduction	4
2.	Énoncé des travaux	4
2.1	Solution proposée	4
2.2	Hypothèses et contraintes	4
2.3	Biens livrables du projet	4
3.	Gestion et suivi de l'avancement	5
3.1	Gestion des exigences	5
3.2	Contrôle de la qualité	5
3.3	Gestion de risque	5
3.4	Gestion de configuration	7
4.	Échéancier du projet	7
5.	Équipe de développement	8
6.	Entente contractuelle proposée	9

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

# Plan de projet

## 1. Introduction

Le plan de projet sert à détailler la proposition soumise afin de répondre à l'appel d'offre. Il définit comment l'exécution du projet se déroulera ainsi que comment il sera surveillé et encadré. Afin d'explorer le marché d'un certain type de jeu, la PME québécoise fictive PolyJeux Inc. désire avoir un prototype. L'équipe de développeurs pour le prototype est constitué d'étudiants en Génie Logiciel à l'école Polytechnique de Montréal, les voici : Simon-Pierre Desjardins, Frédéric Grégoire, Camille Gendreau, Philippe Marcotte, Olivier St-Amour et Ulric Villeneuve. Il sera donc question de décrire non seulement le projet, mais aussi des méthodes qui seront utilisées afin d'assurer la qualité de cette réalisation et d'un échéancier. On y retrouve aussi les rôles de chacun des membres de l'équipe en plus de l'entente contractuelle.

## 2. Énoncé des travaux

### 2.1 Solution proposée

Le projet que nous devons développer possède trois aspects majeurs, un client lourd, un client léger et un serveur. Le client lourd devra posséder toutes les fonctionnalités qui sont actuellement présentes. Parmi ces fonctionnalités il y a un mode édition qui permet à l'utilisateur de construire son propre parcours sur une table, un mode simulation lui permettant de contrôler un robot sur le parcours de son choix et un menu de configuration où il peut ajuster multiple options. Les nouvelles fonctionnalités ajouterons un mode sur le réseau où plusieurs usager pourront rejoindre la même salle de parcours. De plus, ils pourront discuter entre eux à l'aide d'un système de messagerie et voir leur profil à partir du web. Tout comme le client lourd, le client léger permet l'édition du parcours, la possibilité de rejoindre une partie sur le réseau et l'option de clavarder. Il offre également des options tactiles pour iPad. Par contre, le client léger peut uniquement assister à une simulation en mode spectateur. Pour ce projet, un serveur devra être créé pour permettre la connexion entre les multiples clients et s'assurer de la redistribution de l'information à chacun d'entre eux.

### 2.2 Hypothèses et contraintes

Pour ce qui est des contraintes, nous devons fournir les documents répondants à l'appel d'offre avant le 10 février. De plus, le produit final devra être présenté avant le 11 avril. Nous devons compter seulement sur les six membres de notre équipe afin de compléter ce travail. Également, au début du projet, nous n'aurons pas accès à un ordinateur équipé de Mac OS, nous devons donc nous débrouiller à partir d'une machine virtuelle. Il faudra se contenter d'un seul appareil mobile pour le développement sous iOS.

### 2.3 Biens livrables du projet

Nous devons livrer l'ensemble du projet sous deux remises différentes. Lors de la première, le plan de projet, le SRS, la liste d'exigences, le document d'architecture logicielle et le protocole de communication devront être présentés. Ces documents devront être déposés pour le 10 février. Lors de la deuxième remise, le produit final contenant une mise à jour des artefacts remis précédemment, le plan de tests et les résultats de tests est attendu, et ce, pour le 11 avril 2017.

Confidentiel	©Équipe No. 3, 2017	Page 4 de 9
--------------	---------------------	-------------

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

### 3. Gestion et suivi de l'avancement

#### 3.1 Gestion des exigences

Afin de contrôler les changements aux exigences du produit, l'outil *Redmine* sera utilisé. Cela permet de suivre le déroulement de chacune des fonctionnalités et leur complétion.

#### 3.2 Contrôle de la qualité

Tout au long du projet, des tests seront effectués sur les différentes fonctionnalités afin de bien s'assurer qu'elles effectuent les actions attendues. Ainsi, lorsqu'un défaut de fonctionnement sera détecté, il sera possible de le documenter sur *Redmine* afin d'en informer les autres membres de l'équipe et de le corriger par la suite. Avant la remise des livrables, une batterie de tests unitaires sera effectuée afin de tester exhaustivement le logiciel.

#### 3.3 Gestion de risque

La description des risques suit la convention suivante :

- Ampleur : sur une échelle de 1 à 10, 10 étant le risque le plus élevé. Cette analyse est basée sur la probabilité d'occurrence du risque, ainsi que ses impacts.
- Description : une description textuelle du risque ainsi que les problèmes attendus.
- Impact : échelle définissant la portée du risque
  - C – critique (affecte le projet en entier)
  - E – élevé (affecte les fonctionnalités principales du système)
  - M – moyen (devrait être maîtrisable en appliquant une stratégie d'atténuation adéquate)
  - F – faible (l'acceptation du risque est une stratégie envisageable)
- Facteurs : aspects (métriques) du système pouvant être compromis.
- Stratégie de gestion : mesures à prendre afin de gérer le risque.

01 - Remaniement du code				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
5	En remaniant le code déjà existant, nous nous soumettons à la possibilité d'affecter pratiquement, la totalité du code, ce qui mettrait en péril le bon fonctionnement du logiciel. L'impact de ce risque pourrait être catastrophique, mais les chances d'une telle situation est plutôt minime.	C	Cohésion, couplage	Actions correctives

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

02 - Portabilité de l'application				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
3	En traduisant l'application afin de pouvoir la faire fonctionner sur le client léger, nous nous exposons à des risques puisqu'il se peut qu'il doive falloir changer des éléments de la structure ou d'autres complications.	E	Nombre de bogues, cohésion, couplage	Réduction du risque.

03 - Structure non optimale				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
2	Lors de il se peut que nous constatons que la structure actuelle de notre application ne soit pas optimale et qu'on doive la modifier afin d'y remédier.	F	Cohésion, couplage	Actions correctives.

04 - Mauvaise gestion du temps				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
1	On pourrait être tenté de passer trop de temps sur l'esthétique de l'application plutôt que sur l'aspect fonctionnel de celle-ci.	F	Nombre de bogues, temps d'exécution (temps de chargement)	Acceptation du risque

05 - Passage de la physique 2D à 3D				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
7	En implémentant les nouveaux objets de type viaducs, la physique de notre application devra vérifier si les robots peuvent passer en dessous de ceux-ci. Dans le projet précédent, nous avions seulement une physique en deux dimensions, mais afin de bien représenter cette situation, nous devons ajuster cette physique pour qu'elle soit en trois dimensions.	E	Nombre de bogues, nombre de lignes de code, temps d'exécution (temps de chargement)	Réduction du risque - Actions correctives

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

06 - Corruption de la base de données				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
6	Dans le cas d'une corruption de la base de données, les informations liés au système de profil pourraient être affectés et cela peut entraîner de nombreuses complications pour la suite du projet.	E	Nombre de bogues	Réduction du risque - Actions correctives

### 3.4 Gestion de configuration

Au fur et à mesure que les bogues seront découverts, il faudra les ajouter sur la plateforme *Redmine*. Une fois sur celle-ci, tous les membres de l'équipe y auront accès et pourront les revoir, et, par le fait même, les corriger.

Les changements seront aussi documenter sur *Redmine*, puisqu'un historique des versions et des changements est conservé. On peut y voir les changements effectués dans le code. De plus, le nom et la description associés à chacun des "commits" effectués décriront le contexte ainsi que les changements eux-mêmes.

La structure des répertoires devra respecter la structure suivante :

Dans le répertoire "bin", on retrouvera tous les fichiers de type binaires, donc les exécutables, entres autres. Ensuite, dans le répertoire "doc", on retrouvera toute la documentation du projet. Cette documentation sera faite à partir du logiciel Doxygen. Puis, dans le répertoire "ext", on retrouvera les librairies externes utiles au logiciel. Dans le répertoire "src", on retrouvera le code source. Dans le répertoire "res", on retrouvera les ressources supplémentaires nécessaires telles que les modèles de robot, les fichiers de sons, etc. Finalement, dans le répertoire "log", on retrouvera les journaux d'activités du logiciel.

Pour ce qui est des conventions concernant les noms des fichiers, ils ne doivent pas contenir d'espace ou d'accents. De plus, l'extension des fichiers doit être en minuscule. Aussi, en général, les noms de fichier doivent respecter les mêmes conditions que les noms de variables.

## 4. Échéancier du projet

La combinaison des heures-personne des principaux lots ci-dessous totalise 889h-personne. Les 191h/personne restantes sont réservés à des lots de travail essentiel, mais de volumes plus mineurs. Ceci produit donc un total de 1080h-personne.

Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

#### **Livrable 1 (4 semaines) 10 février 2017**

- ❖ Plan de projet
- ❖ SRS
- ❖ Liste d'exigences
- ❖ Document d'architecture logicielle
- ❖ Protocole de communication
- ❖ Prototype de communication client lourd-serveur
- ❖ Prototype de communication client léger-serveur

#### **Livrable 2 (3 semaines) 5 mars 2017**

- ❖ Édition en ligne client-lourd 80h-personne
- ❖ Simulation en ligne client-lourd 80h-personne
- ❖ Clavardage client-lourd 32h-personne
- ❖ Édition sur client-léger 160h-personne
- ❖ Tutoriel client-léger 38h-personne

#### **Livrable 3 (3 semaines) 26 mars 2017**

- ❖ Profils et site web 45h-personne
- ❖ Tutoriels client-lourd 38h-personne
- ❖ Mode course client-lourd 38h-personne
- ❖ Labyrinthe 45h-personne
- ❖ Viaducs 45h-personne
- ❖ Édition en ligne et spectateur simulation client-léger 80h-personne
- ❖ Clavardage client-léger 45h-personne

#### **Livrable 4 (2 semaines) 10 avril 2017**

- ❖ Mini-jeu collection de pièces 38h-personne
- ❖ Mode survie 40h-personne
- ❖ Gestion des droits éditions et synchronisation automatique client-léger 45h-personne
- ❖ Dessin objets client-léger 40h-personne

## **5. Équipe de développement**

L'équipe sera divisée en deux petits groupes de 3 personnes afin d'avoir une équipe pour le client léger et une équipe pour le client lourd ainsi que le serveur.

#### **Client lourd et serveur**

Olivier a de bonnes connaissances en C++ et sera responsable de la mise en place de la base de données et du serveur. Il va également s'assurer de la bonne structure de l'architecture tout au long du projet.

Frédéric sera responsable de l'interface utilisateur du client lourd ainsi que de la page web que nous devons mettre en place. Il a de bonnes connaissances en C# , CSS, HTML et JavaScript. Il aura également à travailler sur les sons et le système de sauvegardes.



Projet Intégrateur 3	Version : 1.2
Plan de projet	Date : 2017-02-01

Camille devra s'occuper majoritairement du serveur tout en supervisant le bon maintien de l'architecture de notre logiciel. Il a une bonne compréhension de la réseautique et du langage C++.

Simon-Pierre possède également une bonne expertise en C++ et sera responsable de travailler sur l'infrastructure du client lourd. Il sera en charge des nouvelles fonctionnalités à intégrer au client lourd.

#### **Client léger**

Philippe a comme responsabilité de devenir un spécialiste en langage Swift. Il va s'occuper de la communication client et serveur tout en travaillant sur l'interface de notre logiciel.

Ulric devra également s'appropriier le langage Swift afin de travailler sur l'interface utilisateur ainsi que pour migrer notre projet du client lourd vers le client léger.

## **6. Entente contractuelle proposée**

Pour ce projet, un contrat de type "livraison clé en main" est plus adéquat puisqu'il nous est demandé de fournir un produit final avec des exigences définies clairement d'avance. Il n'est pas approprié de s'entendre sur un "Contrat à terme - Temps plus frais", puisque ceci requiert de fournir un certain niveau d'effort sur une période définie.. En théorie, cela signifierait que le contrat se termine lors de la livraison et l'acceptation par le client du produit. Cependant, dans notre cas, ceci n'est pas possible, puisque nous sommes fortement contraints par la date de remise du projet.