

UNIVERSITÉ DE SHERBROOKE  
Faculté de génie  
Département de génie électrique et génie informatique

## **RAPPORT APP7**

Informatique : Interfaces graphiques  
APP7

Présenté à  
Charles-Antoine Brunet

Présenté par  
Danick Côté-Martel - cotd2511  
Marc-Antoine Laberge - labm1814  
Jean-Pascal McGee - mcgj2701  
Marc-Éric Pelletier - pelm2528

Sherbrooke - 7 avril 2016

# TABLE DES MATIÈRES

<b>Table des matières</b>	<b>i</b>
<b>Liste des figures</b>	<b>ii</b>
<b>Liste des tableaux</b>	<b>ii</b>
<b>1. Introduction</b>	<b>1</b>
<b>2. Développement</b>	<b>1</b>
2.1 Diagramme de cas d'utilisation	1
2.2 Diagramme de classe de l'interface graphique	2
2.3 Imprimés de l'interface graphique	3
2.4 Explication du but de l'application et du fonctionnement	10
2.5 Explications sur l'ergonomie	11
2.6 Plan de tests de l'interface graphique et de l'application	12
2.6.1 Résultats de trois (3) tests significatifs	14
<b>3. Conclusion</b>	<b>14</b>
<b>4. Références</b>	<b>15</b>

## LISTE DES FIGURES

Figure 1 : Diagramme de cas d'utilisation	1
Figure 2 : Diagramme de classes	2
Figure 3 : Le menu de Sherbrooke Space Agency	3
Figure 4 : Jeu au démarrage de la partie	4
Figure 5 : Progression du jeu	5
Figure 6 : Exemple d'item pour l'avancement du canon du jeu	6
Figure 7 : Progression de l'item du canon du joueur	7
Figure 8 : Fin de la partie du jeu	8
Figure 9 : Bouton retour au menu (démonstration du prototype)	9

## LISTE DES TABLEAUX

Tableau 1 : Plan de tests du menu de <i>Sherbrooke Space Agency</i>	12
Tableau 2 : Plan de tests de l'application de <i>Sherbrooke Space Agency</i>	13

# 1.INTRODUCTION

La problématique de l'APP 7 vise à mettre en pratique les compétences théoriques acquises durant le cours *GIF 250 – Interfaces utilisateurs graphiques* [1]. Le projet développé représente une très proche vraisemblance du projet final de session, *Sherbrooke Space Agency, SSA*, réalisé avec Qt 5.6. Le présent rapport contient la démarche de la solution retenue pour la résolution.

## 2.DÉVELOPPEMENT

### 2.1 DIAGRAMME DE CAS D'UTILISATION

La (figure 1) montre le diagramme de cas d'utilisation de *Sherbrooke Space Agency* :

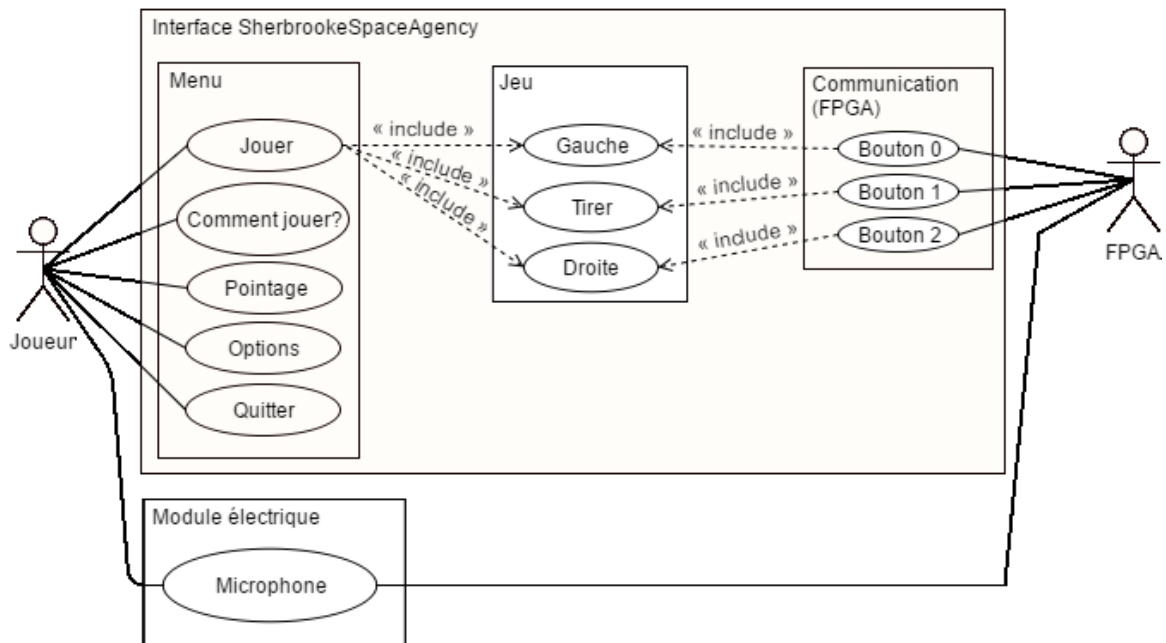


Figure 1 : Diagramme de cas d'utilisation

## 2.2 DIAGRAMME DE CLASSE DE L'INTERFACE GRAPHIQUE

La (figure 2) montre le diagramme de classes de niveau conception de l'interface graphique de *Sherbrooke Space Agency*. Les classes **Qt** sont absentes.

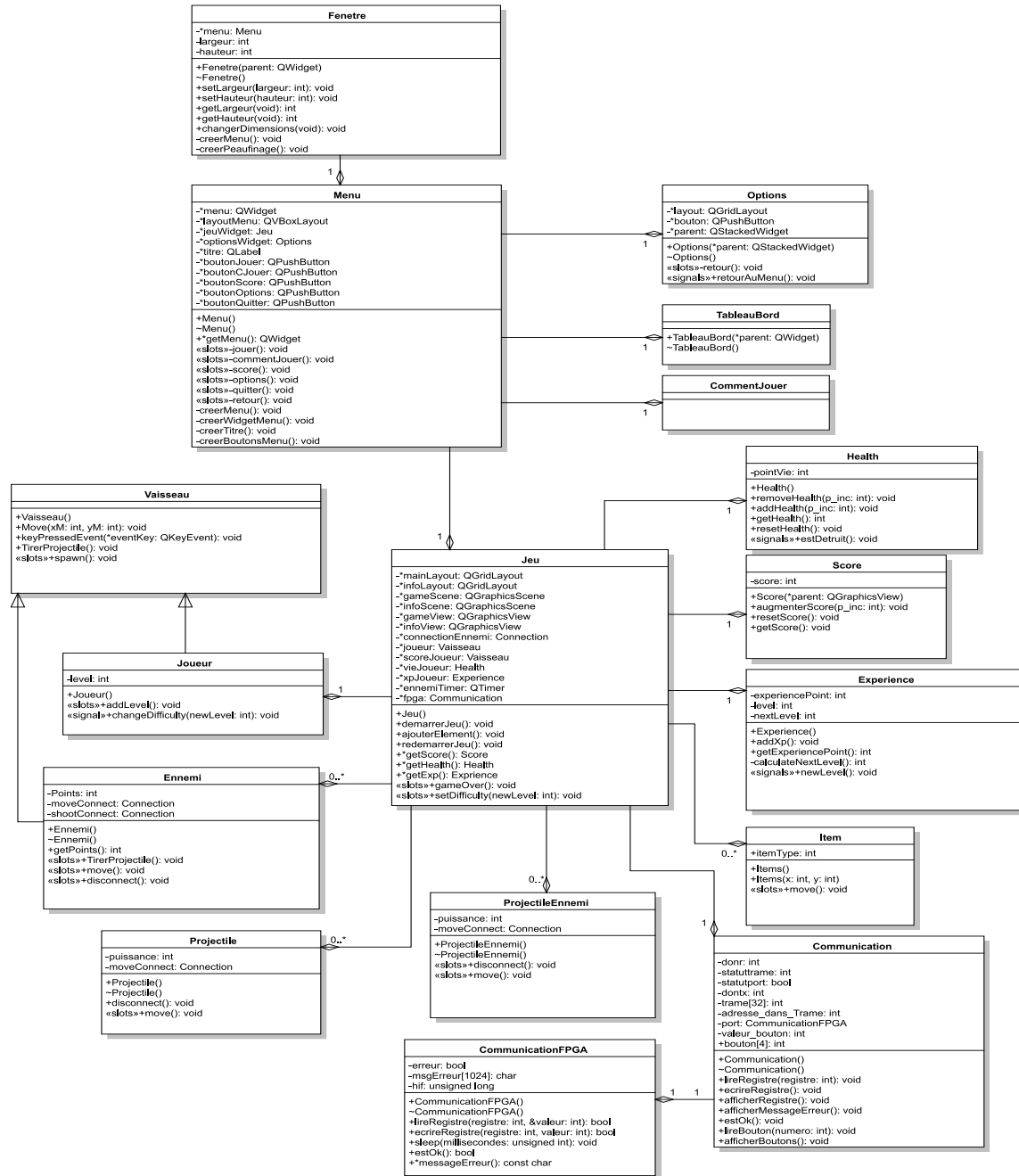


Figure 2 : Diagramme de classes

## 2.3 IMPRIMÉS DE L'INTERFACE GRAPHIQUE

Lors du démarrage de l'application, l'utilisateur va voir un menu avec cinq boutons tels que présenté à la (figure 3).

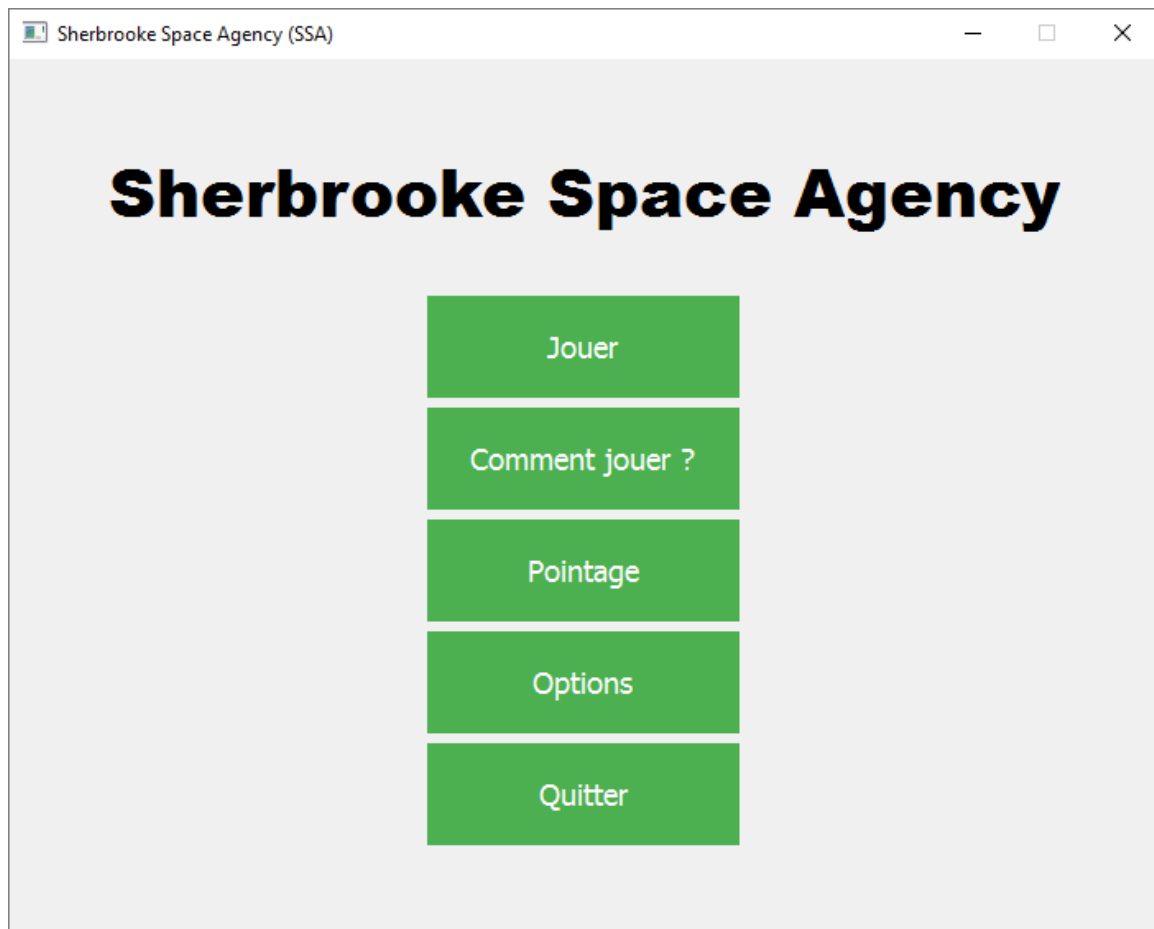


Figure 3 : Le menu de Sherbrooke Space Agency

Une fois dans l'interface de jeu, l'écran est séparé en deux parties, la partie à gauche représente l'aire de jeu, tandis que celle de droite représente des données en rapport avec le jeu. Nous avons une barre de vie au bas, ainsi qu'une barre d'expérience. La barre de vie diminue lorsqu'il y a une collision entre le vaisseau du joueur et un vaisseau ennemi ou un projectile ennemi. L'expérience est calculée en fonction du nombre de vaisseaux ennemis détruits. Une fois la barre d'expérience pleine, elle retombe à zéro et le niveau de difficulté incrémente. Chaque

niveau de difficulté fait apparaître les ennemis à une fréquence de plus en plus grande. Le niveau est affiché dans l'encadré de droit, ainsi que le score et le chronomètre. Le score est calculé à partir du nombre de vaisseaux ennemis détruit, mais il incrémente aussi avec l'aide d'un objet. Le tout est présenté dans l'interface ci-dessous (figure 4).

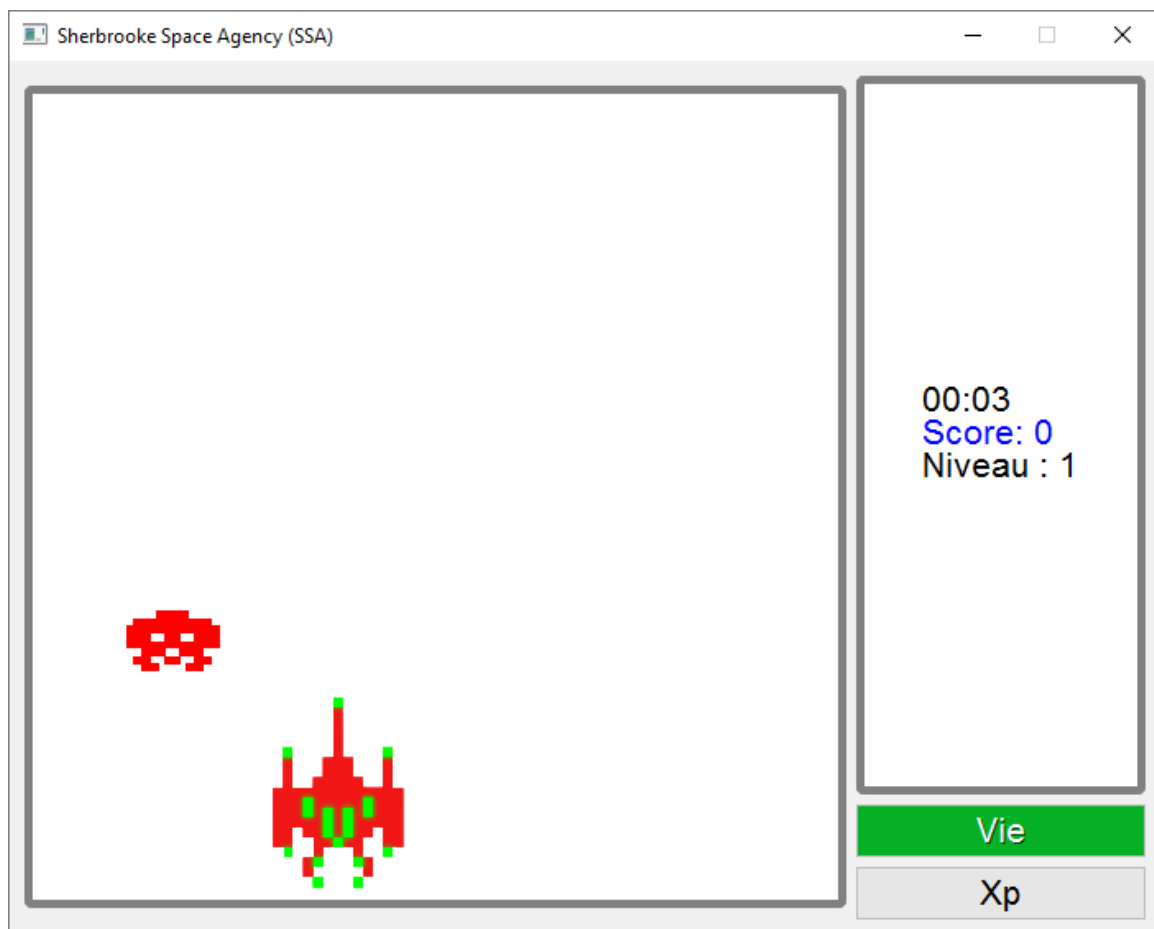


Figure 4 : Jeu au démarrage de la partie

Une progression du déroulement du jeu est présente à la (figure 5).

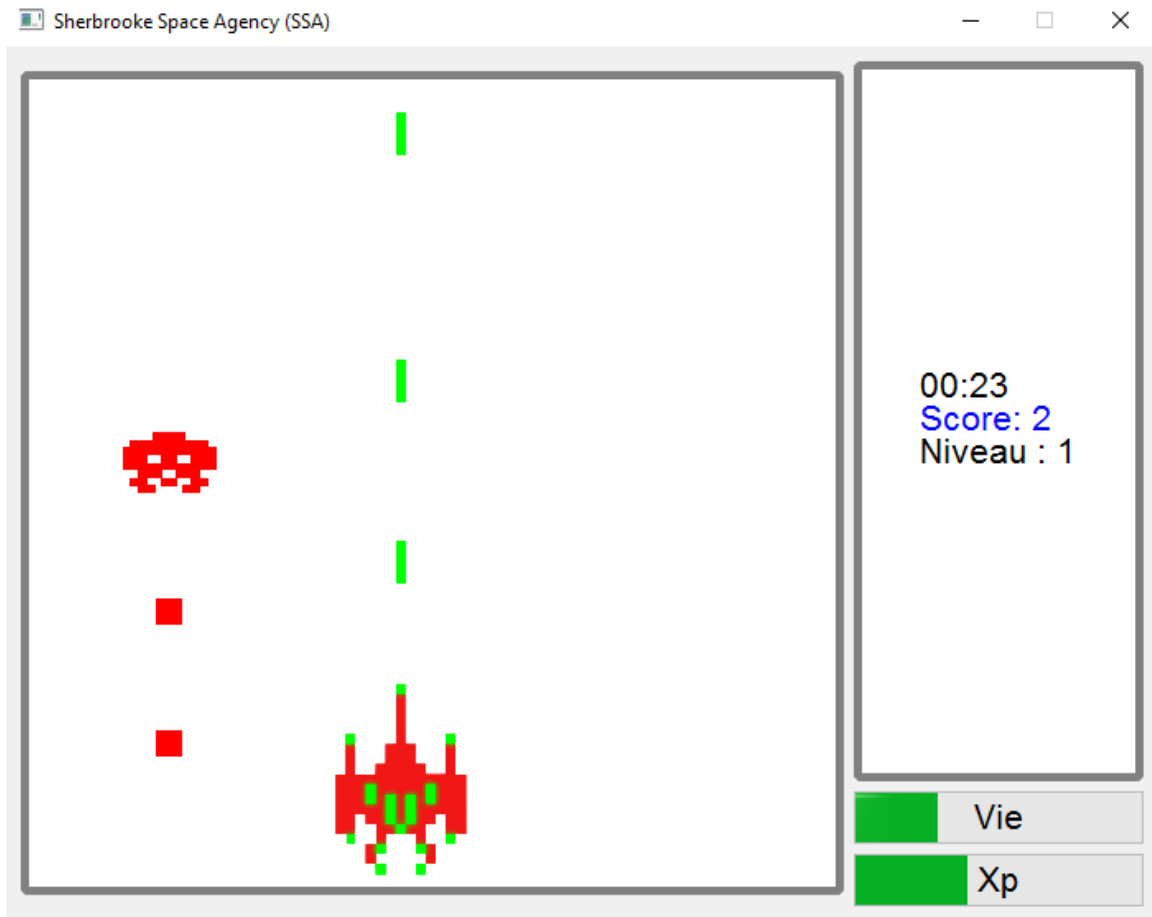


Figure 5 : Progression du jeu

Les objets sont laissés sur le sol rarement lorsqu'un vaisseau ennemi est détruit. Il existe trois sortes d'objets différents, soit un bonus de vie, un canon supplémentaire, ainsi qu'un bonus de score. Ils sont représentés respectivement par un cœur, un missile, et un éclair. Dans la figure suivante (figure 6), l'objet du canon supplémentaire a été laissé par un ennemi, et le joueur doit aller le chercher pour obtenir cette amélioration.



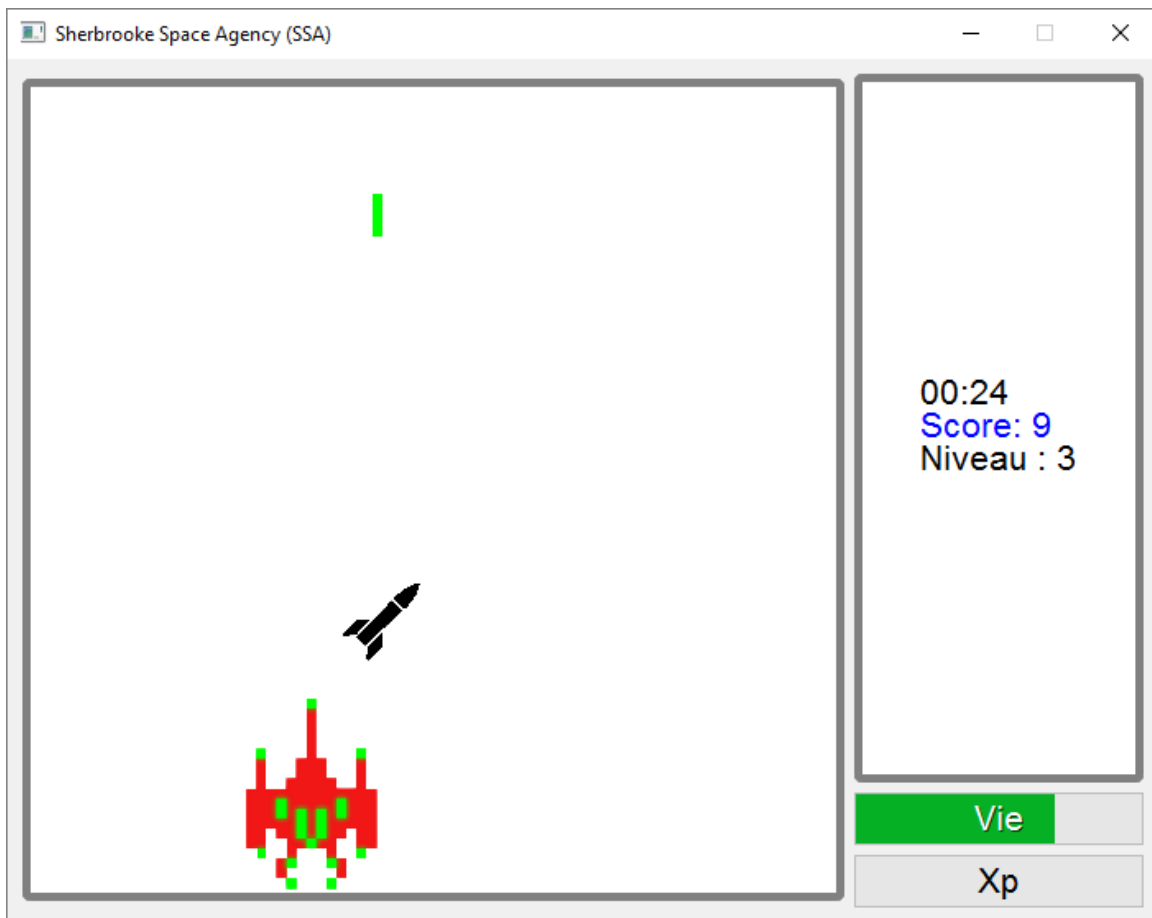
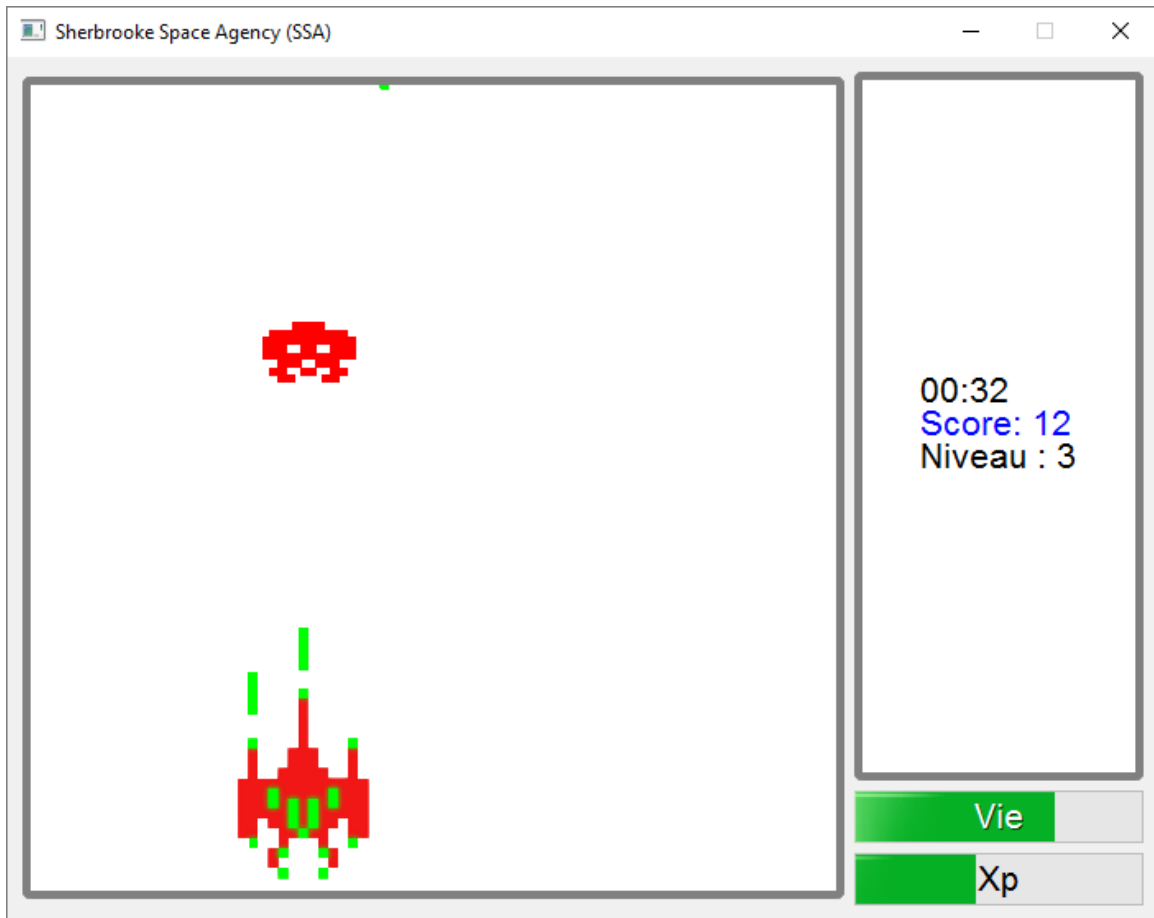


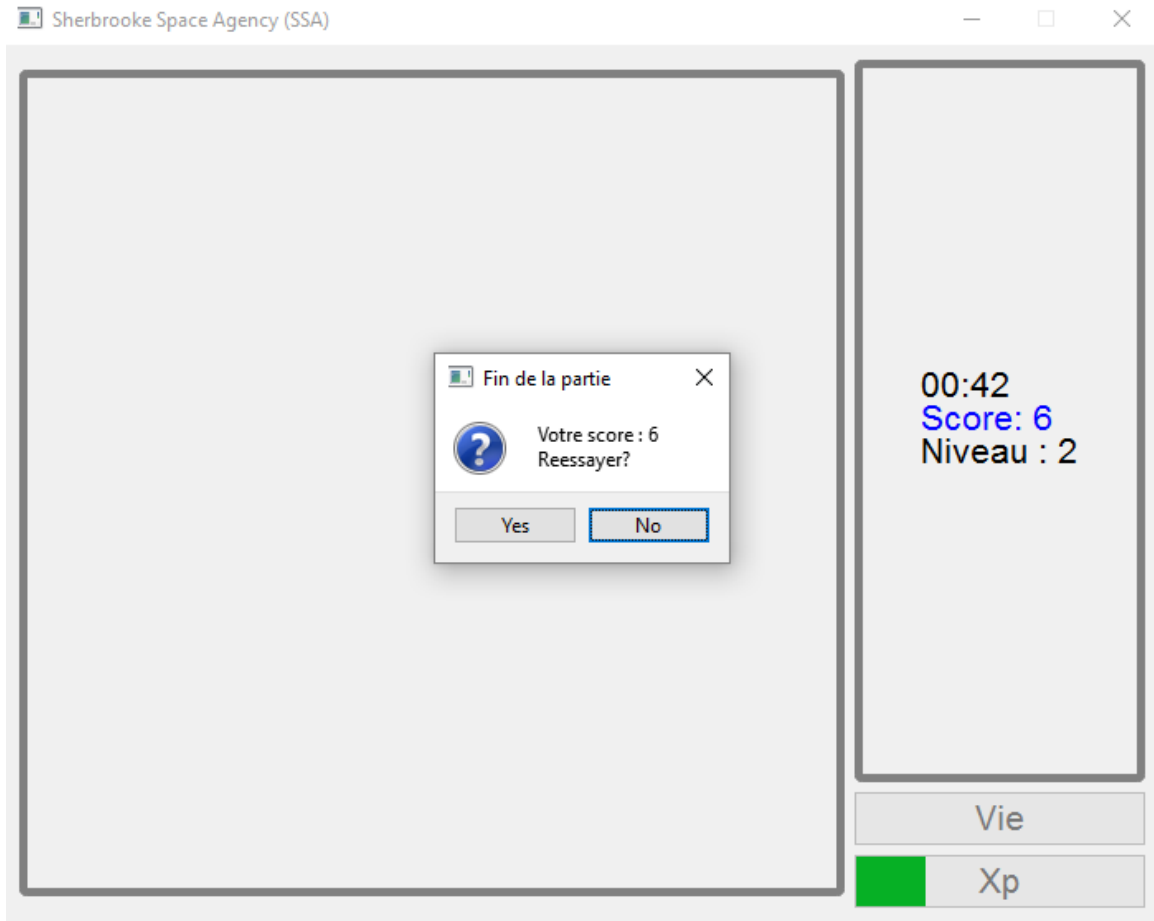
Figure 6 : Exemple d'item pour l'avancement du canon du jeu

Une fois le joueur ayant cette amélioration, il peut tirer un missile de plus, soit dans ce cas deux missiles comme à la figure suivante (figure 7). Si le joueur obtient cet objet une fois de plus, il tirera un troisième missile.



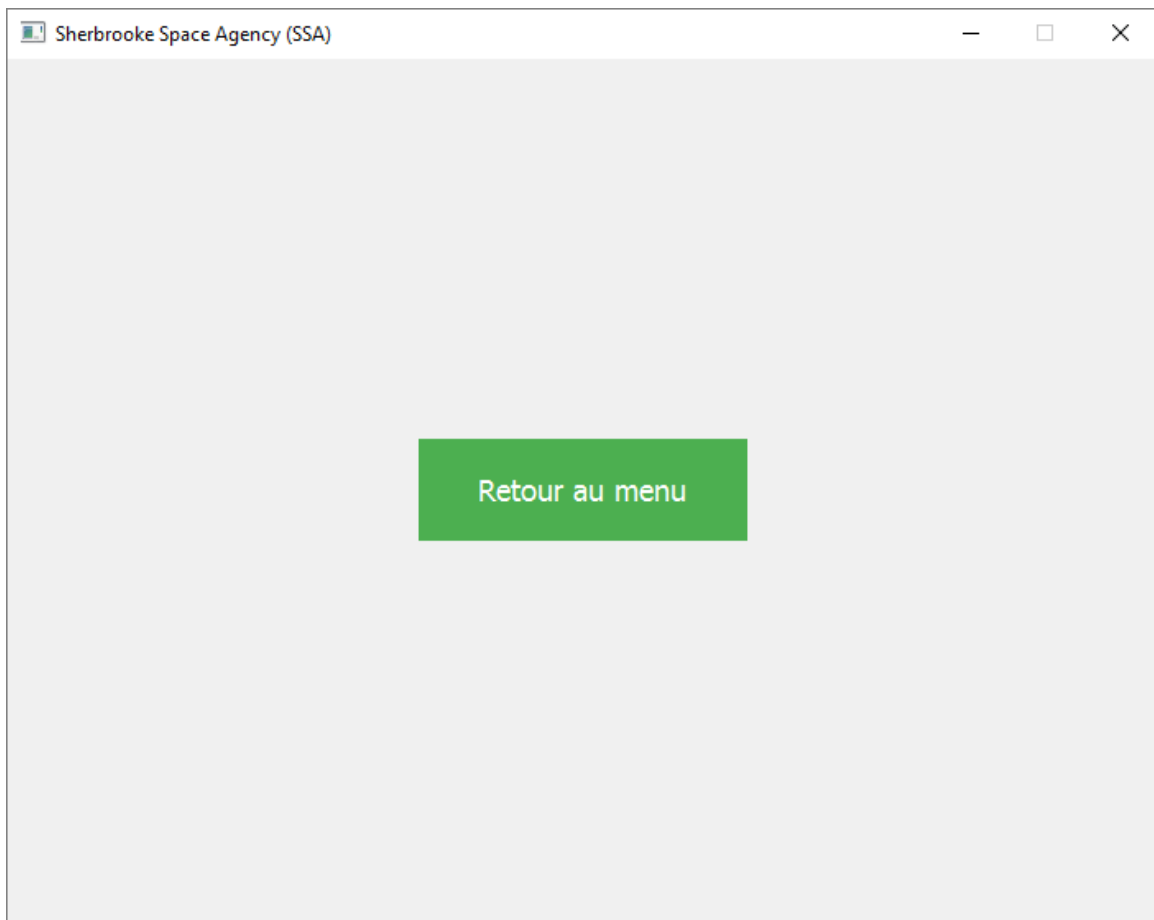
**Figure 7 : Progression de l'item du canon du joueur**

Par la suite, une fois que la barre de vie se vide complètement, le joueur perd la partie. Une nouvelle fenêtre apparaît alors dans le milieu de l'écran en disant le score du joueur et en lui demandant s'il veut réessayer. Si le joueur clique sur oui, la fenêtre de jeu est réinitialisée et le jeu recommence. S'il clique sur non, le joueur retourne à l'interface du menu principal. La figure suivante (figure 8) illustre la boîte de dialogue de fin de la partie.



**Figure 8 : Fin de la partie du jeu**

Une fois de retour au menu, l'utilisateur a une fois de plus un choix de 5 boutons. S'il clique sur le bouton « Comment jouer? », il sera redirigé vers cette fenêtre, pareillement pour les boutons « Pointage » et « Options ». Ces fenêtres n'ont pas encore été implémentées pour la version présente du rapport, et ne font donc rien pour le moment. Cependant, elles contiennent toutes un bouton « Retour au menu » qui permet de revenir au menu. Nous pouvons voir cela à la figure suivante (figure 9).



**Figure 9 : Bouton retour au menu (démonstration du prototype)**

En terminant, le bouton « Quitter » ferme l'application.

## **2.4 EXPLICATION DU BUT DE L'APPLICATION ET DU FONCTIONNEMENT**

Le projet Sherbrooke Space Agency, abrégé SSA, est un jeu de type « abattez-les tous », aussi dit « shoot 'em up » [2] en anglais, contrôlé par la voix. Le joueur contrôle un vaisseau spatial se déplaçant horizontalement dans le bas de l'espace de jeu. Il doit se défendre contre des vaisseaux ennemis, qui défilent du haut vers le bas de l'espace de jeu, en tirant des projectiles vers ces derniers verticalement. De plus, les ennemis peuvent riposter en tirant des missiles vers le bas. Le joueur doit donc éviter les projectiles ainsi que les vaisseaux ennemis et survivre pour passer les différents niveaux. Le joueur peut survivre un nombre fini de collisions, mais n'a qu'une seule vie par niveau alors il doit se déplacer efficacement. Il doit dicter les phonèmes correspondant aux actions « GAUCHE », « DROITE » et « TIRER » soit respectivement le « e » comme dans « le », le « eu » comme dans « peut » et le « i » comme dans « toupie ».

Le joueur possède une barre d'expérience qui incrémente au fur et à mesure qu'il élimine des ennemis. Lorsque la barre est pleine, le niveau de difficulté du jeu augmente, ce qui fait apparaître les ennemis plus vite, puis la barre se vide, et peut être remplie jusqu'à la fin de la partie. Les ennemis laissent tomber des objets une fois de temps en temps qui aident le joueur dans la progression du jeu.

Le but du jeu est de faire le plus de points possible, soit vaincre le plus grand nombre d'ennemis possible. Cela implique de rester en vie le plus longtemps possible pour avoir l'opportunité de vaincre plus d'ennemis, et de ramasser des objets. Un des objets donne dix points au joueur, un autre soigne le joueur en partie, et un dernier le rend plus efficace dans ses combats grâce à un canon supplémentaire. Il faut utiliser le tout le plus stratégiquement possible, car dans les jeux de type « abattez-les tous », le but est d'être le meilleur!

## 2.5 EXPLICATIONS SUR L'ERGONOMIE

Notre application est basée sur les recommandations ergonomiques du *Bureau de la qualité et du support à la déconcentration* (BQSD) de la DSI [1] et de la conception graphique du modèle *C.R.A.P* (Contraste, Répétition, Alignement et Proximité) [2]. Dans le premier document, l'auteur propose plusieurs solutions pour améliorer l'apparence de l'interface, sa navigation ainsi que son utilisation intuitive. Le C.R.A.P soutient les principes de bases du BQSD sous un autre angle.

En ce qui concerne l'apparence, elle est jugée selon la densité de l'affichage, la disposition des éléments dans les fenêtres, les éléments textuels, la typographie, la couleur et les icônes. Ceci dit, la densité de l'affichage répond très bien aux attentes, car le menu présente les boutons principaux, tel que « Jouer », « Comment jouer ? », « Options », « Pointage » et « Quitter ». Dans la version actuelle, seules l'option jouée et l'option quittée sont implémentées. Du coup, dans la fenêtre du jeu, les informations telles que le temps, le pointage du joueur, son expérience et sa vie restante sont jugées nécessaires pour un environnement favorisant l'avancement dans le jeu. D'ailleurs, les éléments textuels ne sont pas très nombreux. La typographie des objets graphiques se limite à 2 polices de caractère, l'Arial Black et Avenir, car l'empatement est absent et l'épaisseur des caractères est constante. Le nombre est limité pour ne pas alourdir la lecture. En ce qui a trait aux images, nous avons jugé pertinent de choisir une image représentant un cœur pour la vie, un éclair pour augmenter le pointage du joueur et un canon pour l'amélioration du nombre canon que le joueur possède. Il est à noter que les bordures des éléments graphiques ont été adaptées avec nuances de gris pour encadrer professionnellement l'action du jeu ainsi que ses informations telles que le minuteur, le pointage et le niveau du joueur. Il est à noter que pour l'instant la barre de vie et la barre d'expérience n'ont pas été modifiées, mais en toute cohérence, elles devront être uniformes aux bordures grises des autres éléments pour être cohérent.

Ensuite, la navigation intra-application choisie dans le projet est la conversation guidée, c'est-à-dire que l'utilisateur est complètement guidé et assisté tout au long de l'application. La chronologie des boutons dans le menu et le choix de l'utilisation est prédéfinie pour gagner en efficacité et se concentrer sur l'objectif final de jouer à Sherbrooke Space Agency. De plus, la profondeur de l'application est limitée à deux niveaux, ce qui est parfait pour ne pas perdre l'utilisateur. Il peut donc facilement naviguer d'une fenêtre à une autre.

Afin d'évaluer si l'interface est intuitive, l'auteur explique qu'il est important de respecter les principes de saisies et le choix des actions associés aux objets. Selon l'application, le joueur doit faire des déplacements latéraux à gauche ou à droite et tirer en avant de lui. Le choix des touches de flèches directionnelles est le plus approprié dans notre situation. D'ailleurs, dans une prochaine version, les raccourcis tels que « W », « A », « S » et « D » seront personnalisables dans les options, car c'est le cas de plusieurs jeux d'ordinateur international. En ce qui concerne le menu, bien entendu, pour quitter l'application, le raccourci « CTRL + Q » a été préconisé.

En somme, tous les éléments proposés par le BQSD viennent renforcer le côté ludique du jeu. La disposition des éléments tels que présenté dans la section 2.3 permet de faciliter et renforcer la communication entre l'application et le client et d'en faire un usage intuitif.

## 2.6 PLAN DE TESTS DE L'INTERFACE GRAPHIQUE ET DE L'APPLICATION

Les tableaux 1 et 2 montrent le plan de tests soumis au programme Sherbrooke Space Agency. Pour une meilleure présentation des résultats, nous divisons selon deux plans de tests, soit le premier tableau relié au menu de l'interface graphique et le deuxième au jeu applicatif.

**Tableau 1 : Plan de tests du menu de *Sherbrooke Space Agency***

Entrée	Objet testé	Observations attendues	Observations obtenues
Clic (menu > jouer)	Bouton Jouer	Affichage du jeu	Réussi
Clic (menu > comment jouer)	Bouton Comment Jouer ?	Affichage des instructions du jeu	Aucune action (pas implémentée)
Clic (menu > pointage)	Bouton Score	Affichage du tableau de pointage	Aucune action (pas implémentée)
Clic (menu > options)	Bouton Options	Affichage des options	Aucune action (pas implémentée)
Clic (menu > quitter)	Bouton Quitter	Quitter l'application	Réussi
Clic	Bouton Retour au menu	Affichage du menu + Arrêt du jeu si on était dans le jeu	Réussi

**Tableau 2 : Plan de tests de l'application de *Sherbrooke Space Agency***

<b>Entrée</b>	<b>Objet testé</b>	<b>Observations attendues</b>	<b>Observations obtenues</b>
Touche (flèche gauche)	Joueur	Déplacement à gauche du joueur	Réussi
Touche (flèche droite)	Joueur	Déplacement à droite du joueur	Réussi
Touche (flèche haut)	Joueur/Projectile	Projectile d'un joueur	Réussi
Touche (flèche haut enfoncé)	Joueur/Projectile	Affichage de plusieurs projectiles	Échoue : Le jeu ralenti Solution : limiter le nombre de projectiles par secondes
Apparition d'un ennemi	Ennemi	Affichage d'un projectile ennemi	Réussi
Projectile d'un ennemi	Projectile Ennemi	Affichage d'un projectile ennemi	Réussi
Collision entre l'ennemi et le joueur	Vie du joueur	Vie du joueur diminué et destruction de l'ennemi	Réussi
Focus de l'objet du joueur	Fenêtre/Jeu/Joueur/	Vaisseau toujours contrôlable	Échoue : focus non joueur Solution temporaire : Cliquer sur le joueur et focus revient
Destruction d'un ennemi	Ennemi	Ennemi détruit et la mémoire est libérée comme prévue	Réussi
Fin du jeu (mort du joueur)	Dialogue (de fin de jeu)	Redémarrer ou retourner au menu	Réussi
Incrémentation du pointage du joueur	Joueur/Score	Incrémentation du niveau du joueur ainsi que son expérience	Réussi
Quantité d'expérience suffisante	Joueur/Expérience	Incrémentation du niveau et de la difficulté	Réussi
Bouton (carte FPGA)	Communication	Déplacement du joueur selon les boutons	Réussi



### **2.6.1 RÉSULTATS DE TROIS (3) TESTS SIGNIFICATIFS**

Les résultats suivants montrent que le programme Sherbrooke Space Agency fonctionne bien.

1. La gestion des boutons du menu : le programme Sherbrooke Space Agency affiche les bons éléments graphiques et retourne au menu si le bouton retour est appuyé et ferme le jeu s'il était dans l'objet Jeu.
2. Collisions des ennemis ou des joueurs avec les projectiles : le programme Sherbrooke Space Agency détruit les ennemis lors des projectiles reçus par le joueur et ce dernier diminue sa vie jusqu'à mourir lorsque sa vie tombe à zéro.
3. La vitesse des projectiles : même si le jeu démontrait une lacune de la gestion des éléments du jeu, il a été corrigé en limitant la cadence du nombre de projectiles tirés.

## **3. CONCLUSION**

La réalisation de notre interface graphique et de son ergonomie est terminée et toutes les classes sont fonctionnelles. Des ajouts supplémentaires tels que des niveaux de compétences, un tableau de bord des meilleurs pointages de chaque joueur et de nouvelles menaces ennemis seront implémentés pour le projet de session.

## 4. RÉFÉRENCES

- [1] C.-A. Brunet, «Département de génie électrique et de génier informatique,» Université de Sherbrooke, 16 Mars 2016. [En ligne]. Available: <http://www.gel.usherbrooke.ca/s2/>. [Accès le 28 Mars 2016].
- [2] Wikipedia, «Shoot 'em up,» 04 03 2016. [En ligne]. Available: [https://fr.wikipedia.org/wiki/Shoot\\_%27em\\_up](https://fr.wikipedia.org/wiki/Shoot_%27em_up). [Accès le 15 03 2016].
- [3] C. Ratier, «GUIDE DE RECOMMANDATIONS ERGONOMIQUES,» équipes projet DSI, 19 Avril 2000. [En ligne]. Available: <https://www.dsi.cnrs.fr/methodes/ergonomie/documentation/guidergo.pdf>. [Accès le 3 Avril 2016].
- [4] G. Reynolds, «chapter6\_spread.pdf,» 2008. [En ligne]. Available: [http://www.presentationzen.com/chapter6\\_spread.pdf](http://www.presentationzen.com/chapter6_spread.pdf). [Accès le 3 Avril 2016].