

Penguin Collector

Date : 16.04.2019

I. Description du jeu

Pitch :

Penguin Collector est un jeu de tir en vue isométrique dans lequel vous incarnez un explorateur dont le but est de **former une colonie de manchots**. Pour cela vous devrez naviguer à travers **des archipels générés procéduralement** pour récolter des manchots et les sauver de leurs prédateurs.

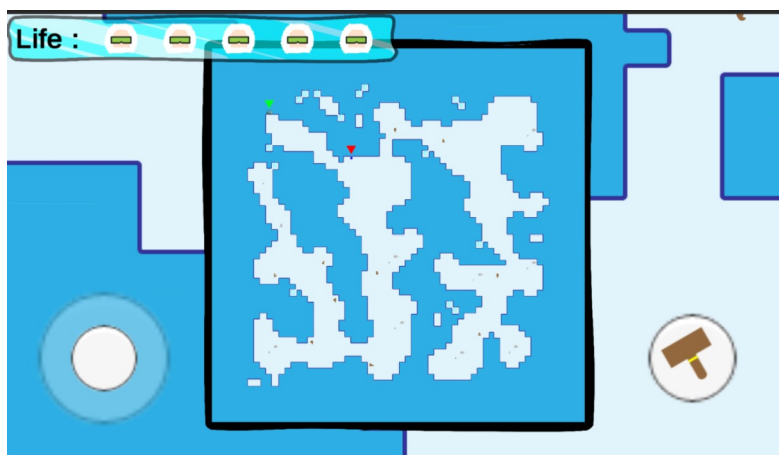
Principe du jeu :

Le joueur contrôle donc un explorateur avec une AK47 pour se défendre face aux prédateurs. Son objectif est de **ramener un maximum de manchots dans son bateau**. Pour cela il doit **explorer l'île** et **libérer les manchots** de l'emprise des ours polaires. Cependant les morses et les ours polaires cherchent à le tuer. Il a alors la possibilité de **se défendre** et de leurs tirer dessus avec son AK47. La partie se termine lorsque **le joueur se fait tuer** par les prédateurs.

II .Aperçu du gameplay :

Camera :

- La camera principale est **centrée sur le joueur** et le suit.
- Une **minimap** est visible dans le coin supérieur droit du jeu permettant d'avoir une vue des alentours du personnage.
- Une **carte entière** est aussi disponible permettant de voir l'entièreté de l'archipel. Un **pointeur rouge** indique **la position du joueur** et un **pointeur vert** indique la **position du bateau**.



Personnages :

- Le joueur est un explorateur qui peut se déplacer dans **toutes les directions** sur l'archipel. Il possède **5 points vies** et meurt s'il les perd tous.
- Il possède aussi un AK47 lui permettant de se défendre. Il **tire dans la direction vers laquelle il se dirige** et ne peut se déplacer s'il est en train de tirer. L'AK47 permet au joueur de tirer en rafale un nombre illimité de balles.



Contrôles :Mobile :

- Le **joystick** en bas à gauche **contrôle le déplacement** du personnage et le **bouton** en bas à droite permet de **tirer avec l'arme**.
- La carte est disponible en **appuyant sur la minimap** et se retire lorsque l'on réappuie dessus.

PC :

- Le joueur **se déplace avec WASD** ou les flèches et **tire avec la barre espace**.
- Il peut **afficher/faire disparaître la carte** en cliquant sur la minimap ou avec **la touche M**.
- Il peut **quitter le jeu** avec **la touche Echap**.
- Des contrôles manettes sont aussi disponibles en cas de préférence.

Personnages Non Joueurs

Les Ours Polaires protègent les manchots et vous agressent si vous vous approchez des manchots, si vous libérez les manchots, ils vous poursuivront jusqu'à vous tuer. Ils possèdent deux points de vies.

Les Morses se déplacent aléatoirement dans leurs zones et poursuivent le joueur s'ils le voient. S'ils sortent de leur zone de chasse, ils retournent à leur point de départ et recommencent leurs déplacements aléatoires.

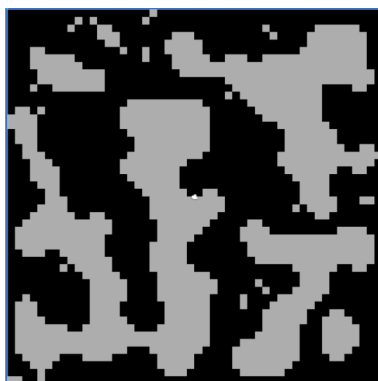
Les Manchots sont prisonniers des ours polaires. Toujours par groupe de 5, ils se libèrent et suivent le joueur si celui ci les touchent. Une fois que le joueur est retourné au bateau, ils rentrent dans le bateau.

III Description de la génération procédurale

A. Génération du terrain

1. Cellular Automata

Pour générer les cases de ma carte, j'utilise le jeu de la vie avec lequel je crée une case si le nombre de cases autour est supérieur ou égal à 5 et avec lequel je détruit une case si le nombre de case autour est égale à 1 ou supérieur ou égal à 4. J'obtiens alors une carte comportant plusieurs régions.



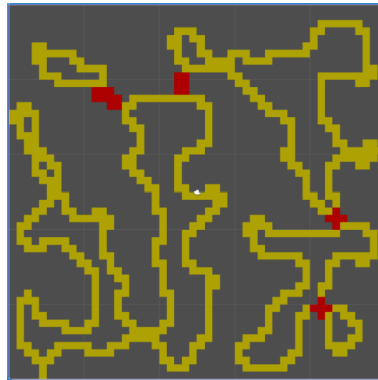
2. Définition des régions

Je recherche à connaître les régions de ma carte, donc j'utilise la technique du Flowfield pour définir à quelle région appartient chaque case et obtenir une liste de ces régions.



3. Connexions des régions proches

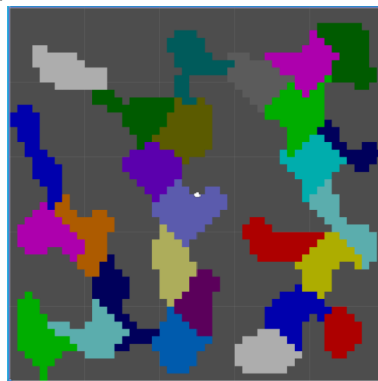
Afin de pouvoir accéder à une grande partie de ma carte je dois relier les plus grandes régions entre elles. Je vais tout d'abord **éliminer les régions trop petites**, puis **définir les bordures** de mes régions. Je vais ensuite chercher les plus petites distances entre les bordures de mes régions et **les relier entre elles**.



B. Génération des ennemies

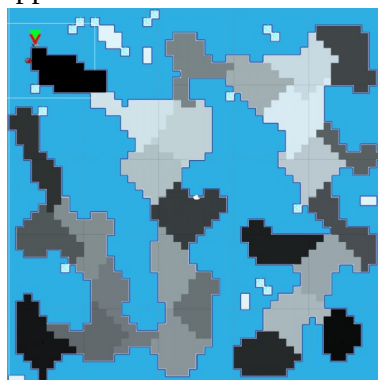
1. Génération des salles

J'ai décidé de générer mes ennemies par salles. Les salles sont des zones générées dans la nouvelle grande région grâce au **Flowfield**. Lorsqu'une zone a assez de case pour créer une salle ou qu'elle se confronte à une bordure, une nouvelle salle se génère. **Les salles trop petites sont ensuite rassemblées avec une salle voisine.**



2. Organisation des salles

Pour définir où doivent apparaître les ennemis, les salles sont organisées par **ordre d'importance**. En premier les salles issues d'une petite région reliée car elles forment un cul de sac, puis les salles sont organisées par rapport aux nombres de cellules formant leur bordure.



3. Création des personnages

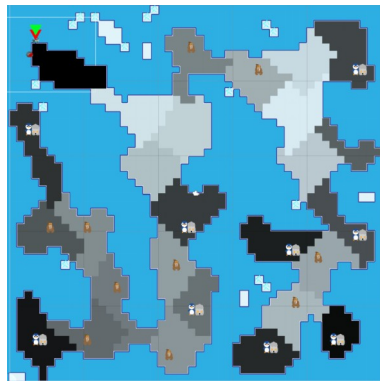
Les personnages sont créés dans les salles les plus importantes qui ne sont pas encore occupées. En premier le joueur, puis les ours polaires et les manchots, puis les morses.

Le joueur est créé sur une bordure pouvant accueillir le bateau.

Les manchots sont créés sur une case entourée de cases et sur l'une de ces cases l'ours polaire les surveillant est créé.

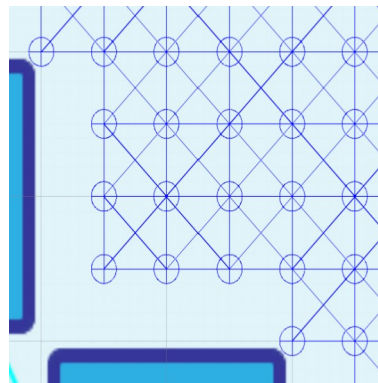
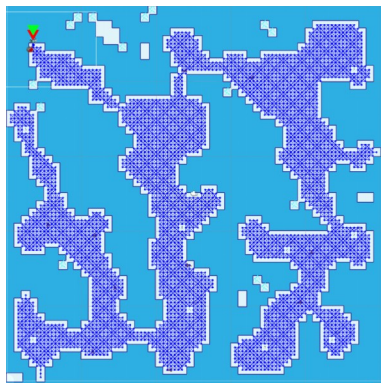
Les morses sont aussi créés sur une case entourée de cases.

Si une salle ne contient pas de case non adjacente à l'eau, cette salle considérée comme occupée et le personnage cherche une nouvelle salle.



III. Description des Intelligences artificielles

A. Représentation du graphe



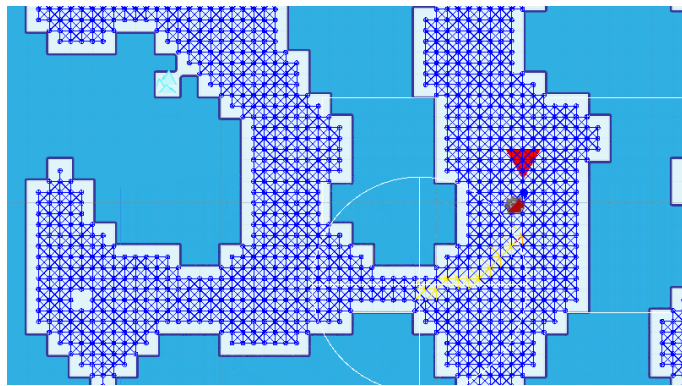
Pour trouver son chemin les intelligences artificielles vont naviguer de nodes en nodes dans un graphique. Les nodes sont générées à partir du tableau obtenu via la génération procédurale. Neuf nodes sont créées sur les bordures et au centre de chaque case de terrain permettant une plus grande précision. Les nodes sont alors connectées à leurs voisines. Deux nodes en diagonales au bord du terrain ne sont pas considérées voisines.

Les nodes sous les manchots prisonniers ne sont pas considérées comme libre évitant aux personnages de rentrer en collision avec les manchots prisonniers.

Si une node fait partie d'une zone occupée, elle va posséder un poids ce qui dissuadera les animaux de traverser une zone occupée par un autre animal.

B. Mode de déplacement

La recherche du chemin des PNJ se fait principalement via l'algorithme A*. Lorsqu'un personnage suit le joueur, un chemin va être recherché depuis le joueur. L'algorithme va prendre la node la plus proche du joueur, regarder ses voisins et les trier en fonction du poids pour les atteindre et de la distance jusqu'au personnage. L'algorithme va alors naviguer de nodes en nodes jusqu'à la node la plus proche du personnage. Une fois trouvée, le personnage va suivre les nodes par rapport aux nodes desquelles elles proviennent.

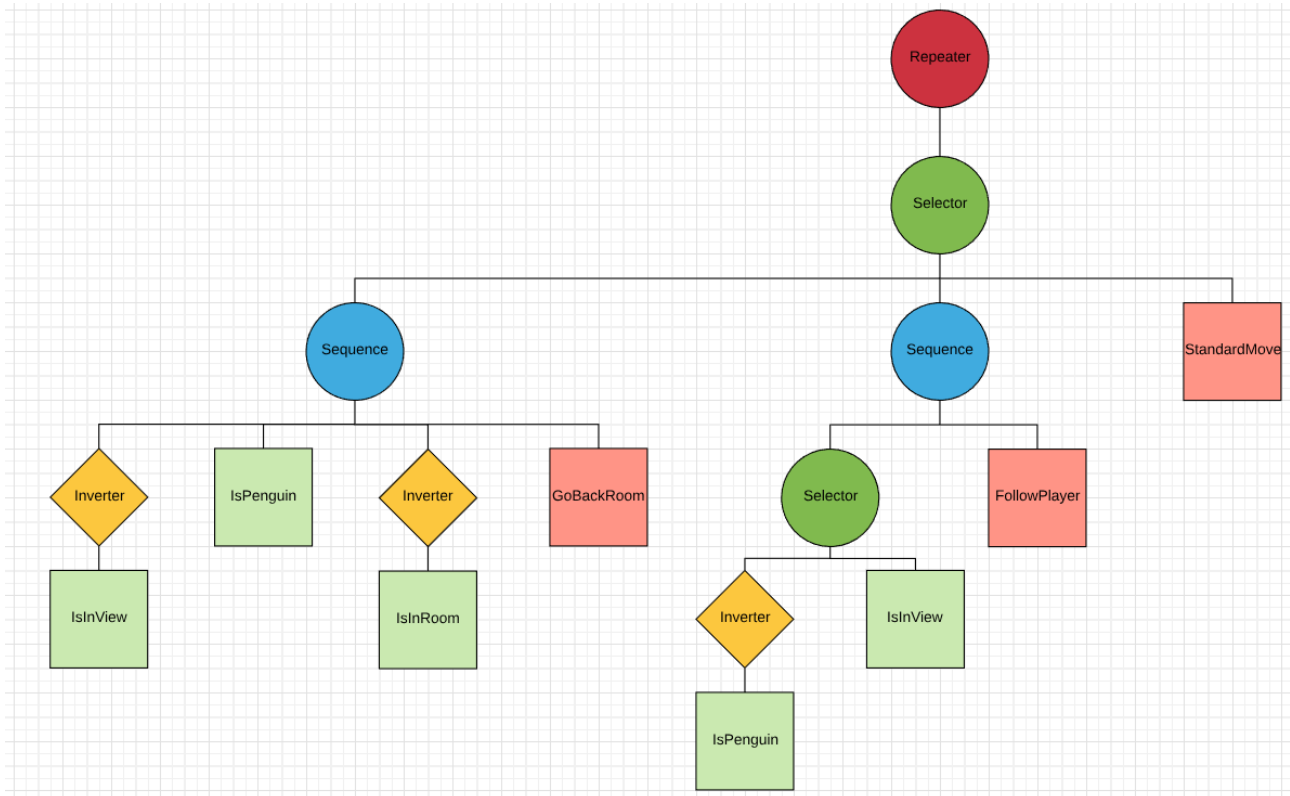


Le déplacement des manchots se fait avec le **Steering Behavior**. Il possède d'abord un **Seek Behavior** pour suivre le joueur, c'est à dire que les manchots vont avoir une vitesse en direction du joueur jusqu'à ce qu'ils soient assez proches. Ils ont aussi un **Separate Behavior** pour les écarter les uns des autres et un **Flee Behavior** les écartant des ennemis et du joueur.

C. Décision des Intelligences Artificielles

Les prédateurs, ours et morses, utilisent un **Behavior Tree** pour savoir ce qu'ils doivent faire.

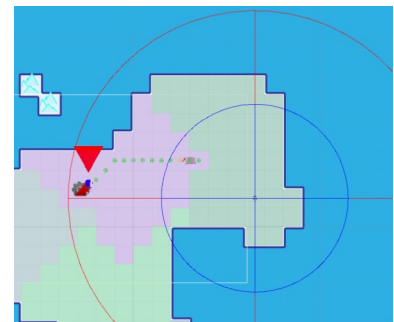
Les **ours polaires** vont d'abord vérifier s'ils doivent **rentrer chez eux** en regardant si le joueur n'est pas proche des manchots, si les manchots sont toujours emprisonnés, et s'ils ne sont pas chez eux. Ensuite ils vérifient s'ils doivent **suivre le joueur** si les manchots sont là et si le joueur est proche des manchots. Sinon ils vont **tourner autour des manchots**.



Standard Move

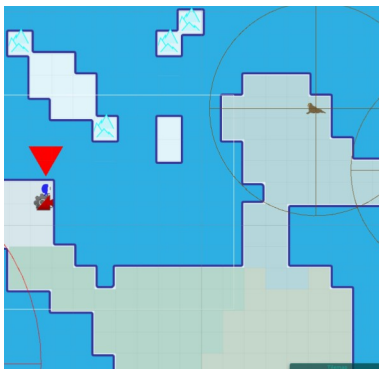
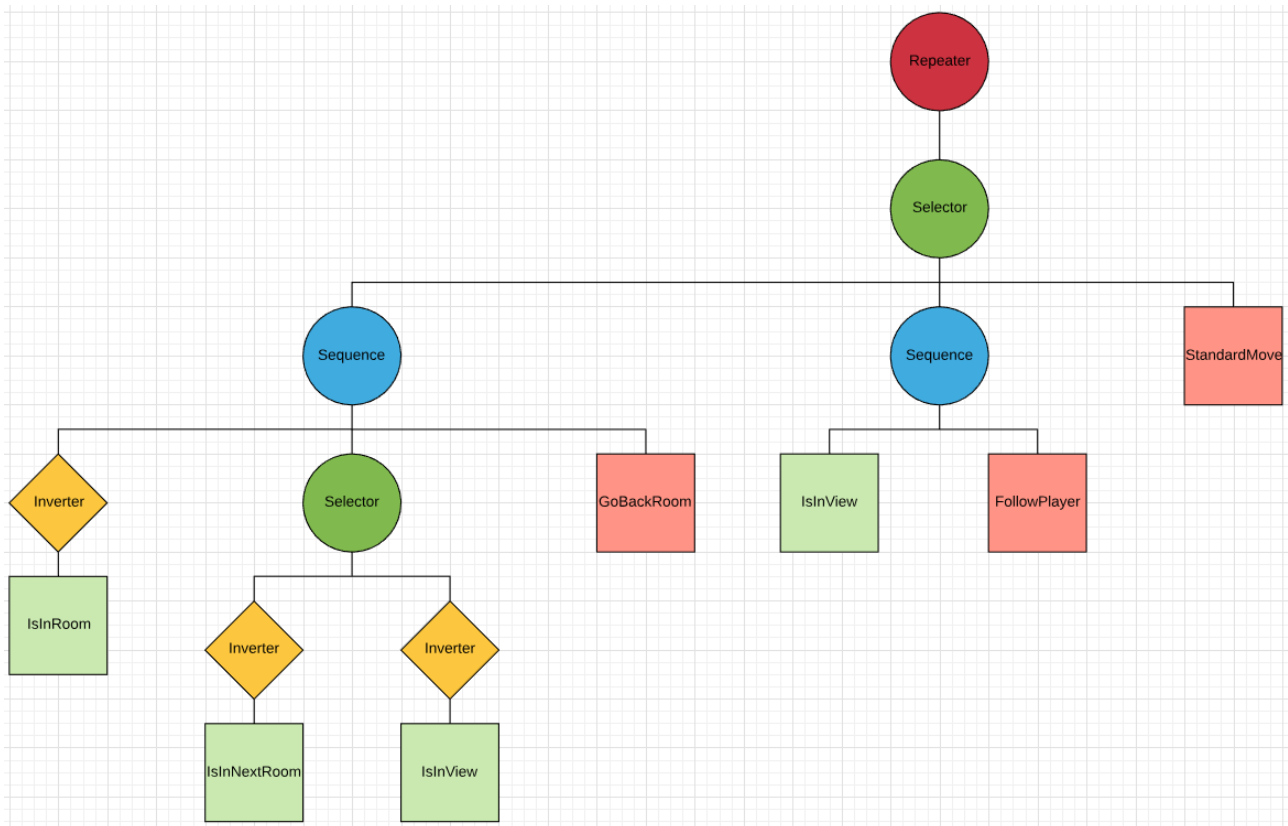


GoBackRoom



FollowPlayer

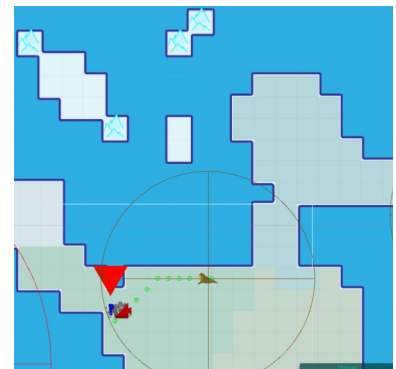
Les **morses** vont d'abord vérifier s'ils doivent **retourner dans leur zone** s'ils n'y sont pas, s'ils ne sont pas dans une zone adjacente ou qu'ils n'ont pas le joueur en vue. Ensuite ils vérifient s'ils doivent **suivre le joueur** s'il le voient, sinon ils vont **se déplacer aléatoirement**.



Standard Move

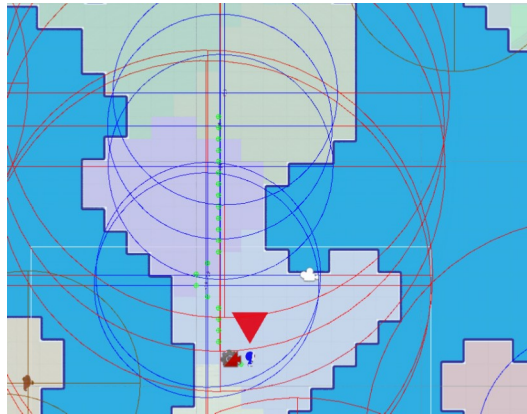
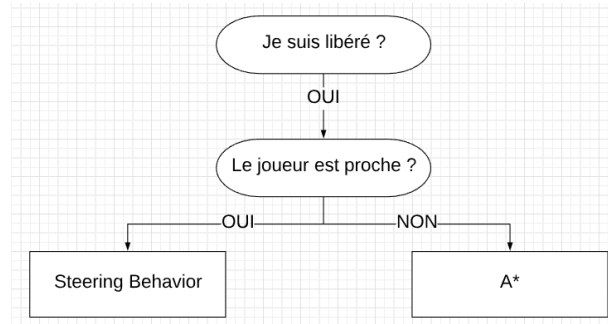


GoBackRoom



FollowPlayer

Les manchots utilisent un **Decision Tree** pour savoir s'ils sont proches ou pas du joueur. S'ils sont proches, ils **suivent le joueur** en avançant dans sa direction. S'ils sont loin, ils **recherchent un chemin** jusqu'au joueur.



Point Vert recherche le chemin vers le joueur car le manchot est trop loin

IV. Conclusion

Améliorations possibles :

- Ajout d'arme plus difficile
- Augmentation de la difficulté au fur et à mesure
- Afficher lorsque le niveau est totalement terminé
- Afficher emplacement des manchots
- Ajout de paramètres pour accessibilité (modification des commandes, ...)

Impressions personnelles :

Grâce à ce projet, j'ai appris la génération procédurale d'un terrain, la mise en place et l'utilisation de l'algorithme A* et les arbres de décisions pour les intelligences artificielles.

Je suis assez satisfait de ce projet et n'hésiterai pas à l'améliorer dans les prochaines semaines.

Crédits :

Asset :

- Ocean Sound : Koenig, M. (2019). *Crisp Ocean Waves Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/1936-Crisp-Ocean-Waves.html> [Accessed 16 Apr. 2019].
- Bear Sound : Koenig, M. (2019). *Devil Dog Crazy Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/1523-Devil-Dog-Crazy.html> [Accessed 16 Apr. 2019].
- Penguin Sound : Koenig, M. (2019). *Guinea Pig Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/2144-Guinea-Pig.html> [Accessed 16 Apr. 2019].
- Gun Sound : Koenig, M. (2019). *Gun Loud Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/1855-Gun-Loud.html> [Accessed 16 Apr. 2019].
- Walrus Sound : Koenig, M. (2019). *Lion Growling Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/1483-Lion-Growling.html> [Accessed 16 Apr. 2019].
- Hurt Sound : Koenig, M. (2019). *Pain Sounds | Effects | Sound Bites | Sound Clips from SoundBible.com*. [online] Soundbible.com. Available at: <http://soundbible.com/1454-Pain.html> [Accessed 16 Apr. 2019].
- Music : Free Music Archive. (2017). *Ice Cream with you by Loyalty Freak Music*. [online] Available at: http://freemusicarchive.org/music/Loyalty_Freak_Music/POSITIVE_ATTITUDE_-_05_Ice_Cream_with_you [Accessed 16 Apr. 2019].
- Boat Sprite : OpenGameArt.org. (2019). *2d boat*. [online] Available at: <https://opengameart.org/content/2d-boat> [Accessed 15 Apr. 2019].
- Snow Font : Dafontfree.net. (2019). *Download free Ice-&Snow Normal font | dafontfree.net*. [online] Available at: <https://www.dafontfree.net/freefonts-ice-snow-f109221.htm> [Accessed 16 Apr. 2019].

Outils:

- Illustrator CC 2018, Photofiltre et <https://fontmemes.com> pour les polices de textes
- Unity 2018.3.8f1

Personne:

Je remercie ma famille et mes amis d'avoir tester le jeu pour me donner leurs impressions.

Je remercie aussi Nicolas Schneider et Elias Farhan et les étudiants Games Programming première et seconde année pour m'avoir donné de nombreux conseils sur le développement.