

Space Encoders:
Un serious game pour apprendre
l'informatique

Marvin FOURASTIE

Juin 2019

Projet de Bachelor
UNIVERSITÉ DE GENÈVE

RÉSUMÉ

Aujourd'hui, l'apprentissage de l'informatique devient de plus en plus populaire dans les écoles dès le plus jeune âge. Il existe de nombreuses applications pour apprendre ces concepts mais celles-ci sont pour la plupart des jeux de programmation sur tablette ou ordinateur. L'idée ici est d'élaborer un jeu de société qui serait donc un « serious game » qui permettrait à un large public de s'initier ou de s'entraîner à travers certains concepts fondamentaux de l'informatique. Nous présenterons donc les règles de *Space Encoders* et comment ce jeu peut être utilisé pour un but pédagogique.

TABLE DES MATIÈRES

1	INTRODUCTION	1
2	ETAT DE L'ART	3
2.1	Définir le « serious game » et son champ d'utilisation	3
2.2	Une méthode de conception d'un « serious game » : le « modding »	4
3	PRÉSENTATION ET ÉLABORATION DU JEU ET DE SES RÈGLES	7
3.1	Le choix du jeu <i>Le Ciel Interdit</i> comme point de départ	7
3.2	Élaboration des règles	8
3.2.1	But du jeu	8
3.2.2	Matériel	9
3.2.3	Déroulement du jeu	11
4	L'UTILISATION DE <i>Space Encoders</i> COMME « SERIOUS GAME »	15
4.1	Public et domaine visés	15
4.2	Potentiel de <i>Space Encoders</i>	15
4.3	Améliorations possibles	16
5	CONCLUSION	19
	BIBLIOGRAPHIE	21

1 INTRODUCTION

L'informatique est de plus en plus présente dans l'apprentissage scientifique que ce soit à l'école dès le plus jeune âge ou encore dans les différentes formations universitaires. Les méthodes d'apprentissage utilisées sont souvent sur support numérique (ordinateur, tablette,...) et sont pour la plupart une initiation à la programmation. Dans cette étude, nous nous sommes intéressé aux jeux éducatifs ayant pour thème l'informatique et nous avons remarqué qu'une grande majorité de ces jeux étaient des jeux numérique « de programmation » du type « faire déplacer un robot ». Nous avons donc décidé d'étudier le fait de pouvoir réaliser un jeu de plateau qui aille plus loin dans l'apprentissage des concepts de l'informatique. Pour cela, nous nous sommes posé les problématiques suivantes :

COMMENT UTILISER LE JEU COMME MÉTHODE D'APPRENTISSAGE ?

Comment allier divertissement et pédagogie ? Pour répondre à cela, nous nous sommes intéressé au concept de « serious game » (que l'on peut traduire comme « jeu sérieux ») et de ce qu'il existe déjà sur le marché et dans la littérature.

QUELS SONT LES CONCEPTS IMPORTANTS À ENSEIGNER ?

Algorithmique, types, mémoire, structures de données, ... Le domaine de l'informatique contient un grand nombre de concepts. L'idée est de sélectionner les concepts les plus pertinents à intégrer dans notre jeu.

COMMENT RÉALISER ET UTILISER UN TEL JEU ?

Règles du jeu, utilisation, public visé sont quelques unes des caractéristiques à prendre en compte pour créer un tel jeu. Pour rendre la partie « game design » moins chronophage nous présenteront comment nous avons conçu notre jeu en partant d'un jeu existant.

A travers ces problématiques, nous allons présenter comment la conception de notre jeu *Space Encoders* nous a permis d'étudier plus en profondeur le concept de « serious game » et comment l'utiliser à des fins pédagogiques.

Nous présenteront tout d'abord en première partie un état de l'art qui sera consacré au « serious game » et comment réaliser ce dernier. La deuxième partie est consacrée

1 Introduction

aux règles du jeu et aux choix que nous avons fait pour concevoir ce jeu. Et la troisième partie expliquera comment utiliser ce jeu et comment tirer le maximum du potentiel de *Space Encoders*.

2 ETAT DE L'ART

Cet état de l'art se réfère à deux documents. Le premier présente le « serious game » à travers ses origines et son utilisation actuelle, le second relate une expérience durant laquelle des chercheurs ont réadapté un jeu existant afin de le transformer en « serious game ». Nous utiliserons ces deux documents pour, en premier lieu, discuter de la notion de « serious game » et proposer une définition clair à ce concept, et, en second lieu, présenter comment concevoir un tel jeu.

2.1 DÉFINIR LE « SERIOUS GAME » ET SON CHAMP D'UTILISATION

Tout d'abord, donnons une définition du concept de « serious game ». Les auteurs de [Djaouti et al., 2011] définissent le « serious game » comme : « des jeux qui n'ont pas le divertissement, le plaisir ou l'amusement comme objectif premier ». Cette définition est une définition large et se rapporte à tout type de « serious game ». En effet, le « serious game » n'est pas utilisé que pour l'éducation, il existe d'autres domaines où le « serious game » existe, comme l'entraînement militaire, la publicité ou encore la culture. Dans notre exposé nous allons nous intéresser au jeu éducatif. Dans le cadre de notre projet, nous avons choisi un jeu de plateau. L'article [Djaouti et al., 2011] donne l'exemple d'un jeu « non-digital » et explicitement labellisé comme un « serious game ». Ce jeu est présenté dans le livre *The New Alexandria Simulation : A Serious Game of State and Local Politics* (Jansiewicz, 1973) qui explique comment jouer à un jeu conçu pour enseigner les bases du système politique des États-Unis. L'auteur de ce jeu a gardé un format « non-digital », pensant que seules les interactions humaines pouvaient simuler la complexité de la politique (Jansiewicz, 2011). En réalisant un jeu de plateau, nous nous sommes approché de la vision de Jansiewicz, car les interactions dans notre jeu visent à faciliter l'apprentissage, qui est le but premier de notre « serious game ».

2.2 UNE MÉTHODE DE CONCEPTION D'UN « SERIOUS GAME » : LE « MODDING »

Définissons tout d'abord le concept de « modding ». « Modder » un jeu est le fait de prendre un jeu de base et de le modifier pour en créer un nouveau. Le document [Castronova and Knowles, 2015] présente comment ses auteurs ont transformé le jeu de plateau *CO2* en un « serious game » pour enseigner les options politiques en fonction du réchauffement climatique. Ce papier propose une méthodologie pour créer un tel jeu avec le « modding » et présente les avantages à utiliser cette méthode. Au cours de la conception de notre jeu, nous avons suivi la méthodologie proposée par le papier [Castronova and Knowles, 2015] :

TRAVAIL SUR LA LITTÉRATURE DU JEU

Chercher des jeux qui semblent proches de notre sujet autant au niveau thématique qu'au niveau des mécaniques [Castronova and Knowles, 2015] permet de trouver des principes qui vont s'approcher du but fixé par notre « serious game ». Il faut aussi nous intéresser aux règles, car il faut que celles-ci soient flexibles pour pouvoir les modifier facilement.

TESTER ET ITÉRER

Ce processus de conception est inévitable pour un jeu de plateau tant la jouabilité est difficile à tester sans essayer une maquette du jeu avec les règles qui ont été modifiées. Nous avons ici la principale phase de conception qui consiste à essayer et modifier le jeu de manière itérative jusqu'à trouver une version satisfaisante. Durant cette phase, il convient aussi de se concentrer sur le système, c'est-à-dire comment le jeu va fonctionner. Le système de jeu ne doit pas être trop complexe, il doit être composé de mécaniques simples qui entraîne une machine complexe [Castronova and Knowles, 2015].

Le papier [Castronova and Knowles, 2015] présente aussi d'autres étapes de conception mais nous avons suivi les deux étapes présentées précédemment. Nous pouvons aussi plus s'attarder sur la jouabilité qui est peu souligné par les documents étudiés mais qui est le concept central lorsqu'on crée un jeu, car un jeu trop compliqué ou punitif devient un jeu qui n'est pas jouable. Cette caractéristique est difficilement vérifiable à travers le fait qu'il peut être un facteur parfois subjectif suivant les affinités des utilisateurs vis à vis des différents types de jeu. De ce fait la seule manière de vérifier si un jeu est jouable est d'y jouer avec différents publics et différentes variantes afin de trouver la version la plus adaptée à ce que l'on recherche.

Après avoir présenté la méthodologie nous pouvons nous questionner sur les avan-

2.2 Une méthode de conception d'un « serious game » : le « modding »

tages à utiliser cette méthode. Ici encore le document [Castronova and Knowles, 2015] nous présente des avantages à utiliser le « modding » et particulièrement deux aspects nous ont intéressés :

PAS DE NOUVELLES TECHNOLOGIES À APPRENDRE

Le fait de reprendre un jeu déjà existant permet de ne pas partir d'une page blanche et de ce fait de se concentrer directement sur la modification des règles du jeu et des mécanismes sans devoir en créer des nouvelles. Cet aspect permet à n'importe qui qui n'est pas expert en « game design » de pouvoir concevoir un jeu.

LIMITER LA VISION

Le fait de ne pas avoir à créer de toutes pièces tous les mécanismes du jeu permet de ne pas surcharger le jeu en fonctionnalités. Les risques lorsqu'on ajoute trop de fonctionnalités sont de perdre la jouabilité et de perdre de vue l'objectif premier qui est de concevoir un « serious game ».

Pour tout cela nous avons utilisé le « modding » pour concevoir notre jeu.

3 PRÉSENTATION ET ÉLABORATION DU JEU ET DE SES RÈGLES

Nous avons présenté dans la partie précédente la définition d'un « serious game » et comment l'utiliser, et une méthode de conception d'un jeu qui est le « modding ». Nous verrons dans cette partie comment nous avons utilisé cette méthode pour créer notre jeu *Space Encoders*.

3.1 LE CHOIX DU JEU *Le Ciel Interdit* COMME POINT DE DÉPART

Pour créer notre jeu, nous avons d'abord décidé de chercher un jeu à « modder ». La visée pédagogique de notre jeu nous a conduit à chercher un jeu coopératif, c'est-à-dire un jeu où tous les joueurs cherchent à réussir un objectif mais où ils jouent ensemble. La fin de ce type de jeu contient deux issues : « tout le monde gagne » ou « tout le monde perd ». Ce type de jeu présente de nombreux avantages. D'abord le fait d'avoir des interactions permet aux joueurs d'apprendre des autres car le niveau des joueurs peut être différent, ou encore le fait de jouer en équipe est un plus, car en informatique ou dans d'autres domaines, le travail d'équipe est très présent. Une deuxième caractéristique qui nous a intéressé est d'avoir un jeu où l'on doit construire quelque chose. En effet, le fait d'avoir un jeu où l'on a une phase de construction nous permet de réutiliser les mécanismes pour pouvoir y intégrer l'écriture d'algorithmes. Nous avons donc choisi le jeu *Le Ciel Interdit* qui regroupe ces caractéristiques.

Comme discuté précédemment, *Le Ciel Interdit* est un jeu coopératif dont le but est de construire un circuit électrique afin de faire décoller une fusée. Pour construire le circuit, les joueurs disposent de tuiles servant à ce déplacer (le plateau se construit au fur et à mesure) et d'éléments à placer pour construire le circuit (câbles, paratonnerres, condensateurs). Les joueurs doivent aussi faire face à différents événements

3 Présentation et élaboration du jeu et de ses règles

qu'il faudra gérer à chaque tour du jeu. En partant de cela, nous avons pu créer nos propres règles en s'inspirant de ces mécanismes.



FIGURE 3.1 : Éléments du jeu *Le Ciel Interdit*

Source: <http://www.cocktailgames.com/jeu/le-ciel-interdit/>

3.2 ÉLABORATION DES RÈGLES

Avant d'élaborer le jeu à proprement dit, il a fallu se demander qu'est ce que notre jeu doit faire apprendre. En effet, notre jeu est un « serious game » et de ce fait il faut pouvoir créer un outil pédagogique. Nous avons d'abord pensé que la programmation d'algorithme et la gestion d'une pile étaient des aspects intéressants à enseigner car elles sont des aspects centraux de l'informatique. Nous verrons néanmoins plus tard que ce deuxième aspect s'est effacé pour renforcer le premier aspect. L'écriture d'un programme s'accordait bien avec les tuiles du jeu de base et nous en avons fait le but principal de notre jeu.

3.2.1 BUT DU JEU

Nous avons adapté le but du jeu de *Le Ciel Interdit* pour *Space Encoders* en donnant comme objectif de construire un programme afin de faire décoller une fusée. De par

cet objectif nous avons voulu inciter le joueur à utiliser les structures de contrôles, les variables locales et les autres éléments liés à l'algorithmique. Pour ajouter de l'enjeu, nous voulions ajouter un système de bonus/malus similaire au jeu de base. Pour cela nous avons mis en place une « pile Orage » qui fait intervenir une série de malus au cours du jeu. Le but du jeu de *Space Encoders* est donc de construire un programme pour faire décoller une fusée tout en gérant les événements qui surviennent.

3.2.2 MATÉRIEL

Le jeu de base contient des « cartes Orages », qui sont des malus, des « cartes Équipements », qui sont des bonus, des « cartes Aventurier », qui contiennent les aptitudes de chaque personnage que l'on peut incarner, et des « cartes Plan », qui modulent la difficulté en augmentant la complexité du circuit. Le jeu contient aussi des tuiles pour construire le plateau, des pions qui représentent les joueurs, une échelle qui fait augmenter le nombre de malus par tour, une flèche indiquant le sens du vent et des composantes pour construire le circuit. Nous n'avons pas utilisé tout le matériel, voici la liste de matériel de *Space Encoders* :

- Des « cartes Objectif » : sur chacune de ces cartes, il y a un objectif représentant une fonction à implémenter. Sur chacune des cartes il y a la signature de la fonction et des indications pour réaliser l'objectif
- Des « cartes Orages » : ils représentent les événements à gérer. Ces événements sont prévisibles.
- Des tuiles : ce sont des morceaux de programme qui ont une sémantique particulière. Ils permettent de réaliser les algorithmes nécessaires à la réalisation de l'objectif.
- Une échelle d'orage : Elle indique le nombre d'événements qui survient, celle-ci augmente au fur et à mesure de l'avancée du jeu.
- Des pions : ils représentent les joueurs.
- Des jetons : ils représentent les variables que l'on peut déclarer dans une fonction.
- Une flèche : indique le sens du vent (haut, bas, droite ou gauche).
- Un compteur : indique les points de vie restants. Les points de vie sont globaux à tous les joueurs.

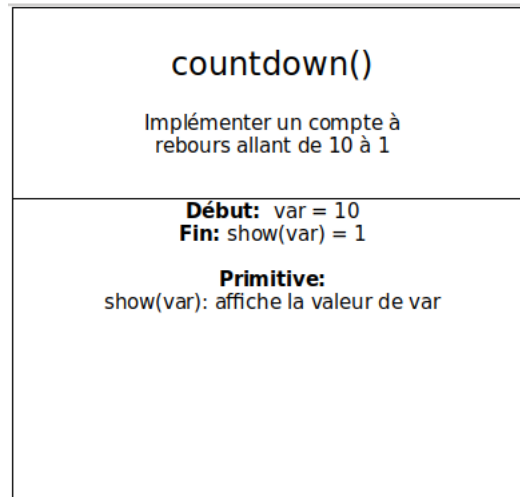


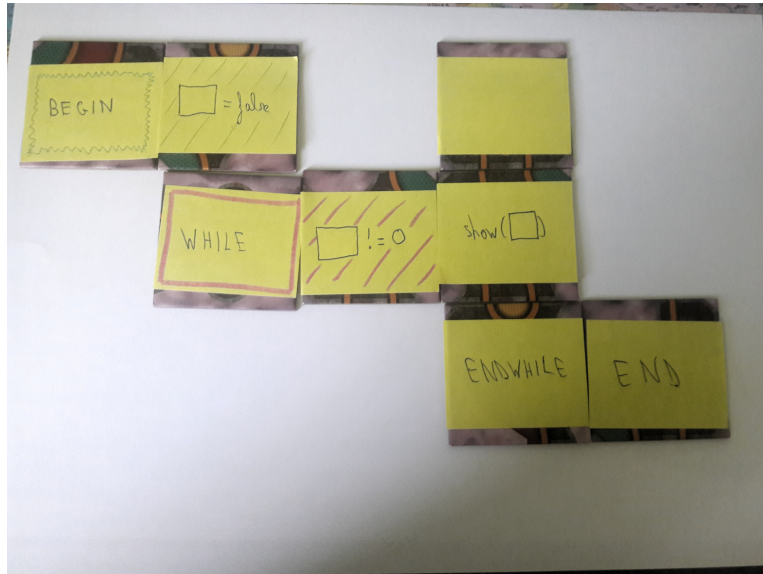
FIGURE 3.2 : Cartes Objectif

Nous avons d'abord pensé à utiliser les « cartes Équipements » mais nous nous sommes rendus compte qu'elles étaient compliquées à exploiter et qu'elles n'étaient pas d'un réel intérêt. À noter que nous n'avons pas spécifié le nombre d'éléments du jeu, car ceux-ci peuvent varier suivant la difficulté et les améliorations possibles du jeu. Pour ce qui est des tuiles, les types de tuiles sont :

- Les début/fin de fonction : BEGIN, END
- Les structures de contrôles : IF, ELSE, ENDIF, WHILE, ENDWHILE, FOR, ENDFOR
- Les variables et opérations sur variables : [] = 10, [] = [] + 1, ...
- Les conditions : [] == length(l), [] = 10 to 0, ...
- Les primitives : show(), fill(), ...
- Les tuiles vides

Les différentes « cartes Orages » sont :

- L'orage se déchaîne (dont une carte qui fait mélanger toutes les « cartes Orages »)
- Le vent tourne
- Bourrasque
- Éclair

FIGURE 3.3 : Maquette de tuiles de *Space Encoders*

3.2.3 DÉROULEMENT DU JEU

Le jeu se joue entre 2 et 4 joueurs, où le nombre d'algorithmes à réaliser est égal au nombre de joueurs moins un. Au début du jeu, les joueurs piochent le nombre de cartes objectifs correspondant (1 pour 2 joueurs, 2 pour 3 joueurs et 3 pour 4 joueurs) et prennent connaissance de l'objectif à réaliser. Ensuite chaque joueur remplit sa main avec 5 tuiles, pour cela le joueur doit piocher 5 fois. L'action de piocher est de retourner 3 tuiles et en prendre une, les tuiles étant placés face cachée sur la table. Dès que tous les joueurs ont rempli leur main, il reste à initialiser la « pile orage » et l'échelle de niveau d'orage. Pour la pile, il faut placer face visible 5 cartes et laisser les autres cartes dans la pioche. Cette pile se lit de gauche à droite. Pour l'échelle de niveau d'orage il suffit de placer le curseur au niveau du nombre de joueurs. Pour ce qui est de l'initialisation de points de vie, son nombre n'est pas encore formellement défini mais des tests ont été réalisés avec 15 et 20 points de vie. Dès que cela est fait, tous les joueurs placent leurs pions sur la case de départ.

La mise en place étant terminée, les joueurs peuvent commencer à jouer. Les joueurs disposent de 4 actions au maximum à chaque tour, ce mécanisme a été gardé du jeu *Le Ciel Interdit*. Dans le jeu de base, les actions consistaient en se déplacer, piocher une tuile, poser une tuile et raccorder (poser une composante pour construire le circuit). La métrique des 4 actions par tour à été gardé mais les actions sont :

3 Présentation et élaboration du jeu et de ses règles



FIGURE 3.4 : Cartes Orages et échelle d'orage

- Se déplacer : un joueur ne peut se déplacer que sur une tuile adjacente en haut, en bas, à gauche ou à droite mais pas en diagonale.
- Piocher une tuile : le piochage d'une tuile se fait en retournant 3 tuiles et en prenant une des trois. Un joueur peut piocher une tuile que s'il possède moins de 5 tuiles dans sa main. Nous avons opté pour ce système de choix pour inciter le joueur à réfléchir sur la tuile qu'il aura le plus besoin pour construire son code.
- Placer une tuile sur le plateau : comme pour le jeu de base, le plateau se construit au fur et à mesure. On ne peut placer une tuile seulement à une position adjacente d'où se trouve notre pion (de manière similaire au déplacement).
- Échanger une tuile : il est possible d'échanger une tuile avec la pioche de tuile, avec un autre joueur ou avec une tuile du plateau. Dans le premier cas, on pioche une tuile et on remet une tuile de notre main dans la pioche, dans le second on échange une tuile contre une autre avec un autre joueur et dans le dernier cas on peut échanger une tuile du plateau avec une de notre main seulement s'il n'y a pas de joueur sur la tuile.

Les joueurs peuvent réaliser au maximum 4 actions parmi celles-ci, dans l'ordre qu'il veulent et peuvent répéter une action autant qu'il le veulent. Quand tous les joueurs ont joué, les événements de la « pile Orage » surviennent, à la base nous avons prévu que ces événements surviennent à chaque tour, mais après de nombreux tests le jeu devenait trop punitif plus le nombre de joueur augmentait. Le nombre d'événements subis est indiqué par l'échelle d'orage. Présentons maintenant les effets des « cartes Orage » :

- L'orage se déchaîne : fait augmenter d'un cran l'échelle d'orage. Si la carte indique de mélanger toutes les cartes, il faut récupérer toutes les cartes, les mélanger et initialiser de nouveau la pile.
- Bourrasque : fait déplacer les joueur d'une case dans le sens indiqué par la flèche. Si un joueur tombe, c'est-à-dire est poussé à un emplacement où il n'y a pas de tuile, chaque joueur tombé fait perdre un point de vie. Un joueur tombé retourne sur la case où il était initialement.
- Le vent tourne : Fait tourner la flèche du vent dans le sens indiqué par la carte.
- Éclair : Fait perdre un point de vie. Nous n'avons pas encore décidé de l'intégration de cette carte dans le jeu pour la principale raison que c'est la seule qui ne peut pas être évitée.

Dès que les joueurs ont subis les « cartes Orage », on enlève un point de vie et on remplit la « pile Orage » en ajoutant les cartes de la pioche à droite de la pile jusqu'à ce qu'il y ait 5 cartes retournées. Les cartes consommées sont remis sous la pioche. L'idée d'enlever un point de vie quand tous les joueurs ont fait leur tour est de faire en sorte que les joueurs réussissent à faire leurs objectifs en un minimum de tours. D'où les points de vie seraient similaires à une « dead line ».

La construction des algorithmes à l'aide des tuiles suit des règles de syntaxe comme tout langage informatique. La construction se fait à la manière du jeu de dominos, à l'exception près que le sens de lecture est unidirectionnel. Nous avons longuement travaillé sur cette syntaxe pour qu'elle soit claire, lisible et facile à comprendre. Listons les règles de syntaxe :

- Tout programme doit commencer par une tuile BEGIN et se terminer par une tuile END.
- Les déclarations de variables se font après une tuile BEGIN ou une autre tuile de déclaration de variable.

3 Présentation et élaboration du jeu et de ses règles

- L'utilisation d'une structure de contrôle se fait dans l'ordre suivant : [FOR, IF ou WHILE], [condition](1 tuile), [primitives ou opérations sur variable](1 ou plusieurs tuiles), [ENDFOR, ENDIF ou ENDWHILE].
- Seules les tuiles vides peuvent être placées n'importe où (tout en respectant l'action « placer une tuile »)
- Il est interdit de placer une tuile qui construit deux chemins possible de lecture.

Les tuiles vides ne représentent pas du code, ce sont juste des plate-formes utiles pour se déplacer. Pour aider, les tuiles ont un code couleur indiquant les compatibilités entre les tuiles. Pour déclarer une variable, il faut poser une tuile de déclaration de variable puis poser sur le [] de la tuile un pion de couleur où chaque couleur est un « nom » de variable.

Tous les joueurs ont gagné si tous les objectifs sont réalisés sans que le nombre de points de vie atteigne 0.

4 L'UTILISATION DE *Space Encoders* COMME « SERIOUS GAME »

4.1 PUBLIC ET DOMAINE VISÉS

Au début de notre projet de « serious game », nous souhaitions faire de *Space Encoders* un jeu qui s'adresserait au plus large public possible. Mais durant l'élaboration du jeu nous avons constaté que notre jeu était bien adapté pour un cadre scolaire pour des personnes ayant peu d'expérience en informatique comme pour des personnes confirmées en informatique. Le facteur du jeu qui nous a fait rendre compte de cela est la vérification du programme, car pour jouer au jeu convenablement il faut comprendre la sémantique de ce que l'on écrit et, par conséquent, il faut une personne sachant programmer pour vérifier le programme écrit. C'est pour cela que ce jeu est parfaitement adapté au cadre scolaire. Mais *Space Encoders* n'est pas un jeu qui est spécialisé pour un cadre scolaire, il peut être intéressant d'y jouer avec un groupe de personne qui contient une personne confirmé en programmation. Dans ce cas la, les autres joueurs pourront en apprendre sur la programmation en simplement une heure de jeu, ceci est la force d'un jeu coopératif, car il permet aux joueurs d'apprendre grâce aux interactions pendant une partie.

4.2 POTENTIEL DE *Space Encoders*

Tout l'enjeu de ce projet était de concevoir un « serious game » à visée pédagogique, de ce fait les éléments et les mécanismes du jeu ont été pensés pour faire apprendre aux joueurs certains concepts de l'informatique.

Le premier concept est l'algorithmique. Tout d'abord on peut citer les tuiles qui permettent au joueurs d'apprendre à utiliser les structures de contrôles, à déclarer et utiliser une variable ou encore à gérer la portée des variables. De plus, le fait d'avoir des tuiles délimitant la portée du programme et des structures utilisées permet l'assimilation des boucles et des tests.

Le second concept est l'abstraction. Le fait de construire un programme pour répondre

à un problème est l'essence même de ce qu'est l'informatique. Les cartes objectifs permettent aux joueurs de réfléchir comme un programmeur, c'est-à-dire penser à ce que j'ai de disponible et ensuite comment répondre au problème avec ce que j'ai. C'est pour cela que nous avons pensé à proposer un large choix de tuile pour laisser la liberté aux joueurs de construire leur algorithme comme ils le souhaitent, mais la présence des points de vie fait que les joueurs doivent utiliser la solution la plus efficace pour réaliser l'objectif. Ceci est aussi une problématique qui est présente lorsqu'on est programmeur.

Le troisième concept est la planification. Anticiper ce qui pourra être un futur problème permet, comme dans de nombreux jeux, de préparer plusieurs tours à l'avance. Même si la planification est présente dans le domaine informatique, c'est un facteur qui est prépondérant dans d'autres domaines.

Le dernier concept est le travail en équipe. Là encore, dans beaucoup de domaines le travail d'équipe est présent. Comme discuté précédemment, dans *Space Encoders* le fait de construire les programmes en équipe permet un partage des connaissances. Un autre facteur intéressant de ce jeu est que les différents objectifs à implémenter peuvent faire varier la difficulté du jeu. Il est possible alors d'utiliser ce jeu pour un public de différents niveaux. De plus nous voulons rendre le jeu extensible en se donnant la possibilité d'ajouter des objectifs au jeu. Pour cela, il suffit de concevoir les tuiles correspondantes aux algorithmes à réaliser.

4.3 AMÉLIORATIONS POSSIBLES

Arrivés vers la fin de notre projet, nous avons pensé aux améliorations que nous pourrions apporter. Tout d'abord, comme nous l'avons souligné dans le chapitre précédent, il nous faudrait définir de manière précise le nombre d'éléments du jeu. En effet, la multiplication des tests du jeu nous permettrait de définir le nombre de « cartes Orage » de chaque type à mettre dans le jeu. Il reste aussi à définir le nombre de points de vie pour une partie, cela demande une analyse plus poussée du jeu, car cela dépend de tous les autres éléments du jeu. Une analyse statistique pourrait aider à définir une mesure pour le nombre de points de vie. *Space Encoders* est encore sous la forme d'une maquette, il serait intéressant de se pencher sur le design visuel du jeu et, pourquoi pas, revoir la nomenclature des cartes.

Il est aussi possible d'exploiter le jeu pour travailler sur différentes chose. Il pourrait être intéressant de proposer un programme pour simuler des parties afin de pouvoir travailler avec des algorithmes d'intelligence artificielle. Cela permettrait à la fois

4.3 Améliorations possibles

de jouer contre une IA mais aussi de pouvoir analyser plus précisément le jeu et de pouvoir apporter les améliorations citées précédemment.

5 CONCLUSION

Tout au long de cette étude, nous avons présenté comment concevoir et exploiter un « serious game » ayant une visée pédagogique pour apprendre l'informatique. Nous avons vu que le « serious game » est un jeu dont le but principal n'est pas le divertissement et que ce format qui est le jeu de plateau est adapté à la visée éducative recherché par notre jeu. A l'issue d'une recherche de littérature, nous avons relevé une méthodologie pour concevoir un tel jeu : le « modding », qui est le fait de prendre un jeu existant et d'en changer les mécanisme pour créer un autre jeu.

Partant de cette méthodologie, nous avons sélectionné un jeu coopératif qui est *Le Ciel Interdit* et avons modifié les règles et les mécanismes pour créer *Space Encoders*. Ce jeu permet au joueur de s'initier à l'écriture d'algorithmes par le biais d'objectifs à réaliser en équipe. Le format de ce jeu est propice au milieu scolaire car il fait appel à de nombreuses compétences telles que l'algorithmique, l'abstraction et la planification.

Ce jeu est encore une maquette et de ce fait il peut ouvrir à de nombreuses améliorations que ce soit au niveau du game design qu'au niveau de son analyse.

Nous avons démontré au cours de cette étude, que le « serious game » est une alternative intéressante dans l'apprentissage de l'informatique et permet l'enseignement à plusieurs niveaux. De ce fait, *Space Encoders* est un jeu qui mérite une attention particulière car il est rare de trouver un jeu similaire sur le marché.

BIBLIOGRAPHIE

- [Castronova and Knowles, 2015] Castronova, E. and Knowles, I. (2015). Modding board games into serious games : The case of Climate Policy. *International Journal of Serious Games*, 2(3) :22.
- [Djaouti et al., 2011] Djaouti, D., Alvarez, J., Jessel, J.-P., and Rampnoux, O. (2011). Origins of Serious Games. In Ma, M., Oikonomou, A., and Jain, L. C., editors, *Serious Games and Edutainment Applications*, pages 25–43. Springer London, London.