



Laboratoire informatique de l'université du Mans  
Site de LAVAL, CERIU2  
Équipe IEIAH  
52 Rue des Docteurs Calmette et Guérin,  
53000 LAVAL  
Tél : +33 (0)2 43 59 49 01

Institut Universitaire de Technologie de LAVAL  
Département informatique  
52 Rue des Docteurs Calmette et Guérin,  
53000 LAVAL  
Tél : +33 (0)2 43 59 49 09

## **Escape It**

Jeu sérieux pour l'aide à l'apprentissage visuel  
des enfants autistes

Rapport de Stage de fin de cursus  
pour l'obtention du Diplôme Universitaire de Technologie (DUT)  
informatique

Réalisé par  
**Léo MOTTAY**

Du 9 Avril 2018 au 22 Juin 2018

Au sein du LIUM

Soutenu le 26 Juin 2018 à LAVAL

Maître de stage

**Pierre LAFORCADE**

Tuteur IUT

**Clément LABORIE**

Je tiens à remercier tout le corps enseignant de l'IUT de Laval, pour la qualité de leur enseignement et pour m'avoir donné tous les outils nécessaires à la réussite de ce stage.

Je tiens également à remercier Ludovic HAMON pour son encadrement et ses conseils quant à la recherche du stage.

Je tiens particulièrement à remercier Pierre LAFORCADE pour m'avoir accordé sa confiance, pour la qualité de son encadrement, pour la qualité de son sujet de stage, et pour la qualité de l'expérience de travail que j'ai pu avoir avec lui.

Je tiens à remercier Clément LABORIE, Youness LAGHOUAOUTA, Esteban LOISEAU et Sébastien GEORGE pour m'avoir épaulé à leurs façons.

Je tiens à remercier tous les membres du bureau qui m'ont été assignés pour leur accueil et pour avoir grandement facilité mon intégration dans les locaux.

Je tiens enfin à remercier Quentin COULAND et Vincent BETTENFELD pour le soutien moral et le golden pepe.

Le LIUM est le Laboratoire Informatique de l'Université du Maine. Il a environ trente ans et comporte 28 enseignants-chercheurs, 26 doctorants et chercheurs non permanents, le tout dirigé par Sylvain Meignier. Ce laboratoire est divisé en deux équipes: La première située sur Le Mans nommée LST(Language and Speech Technology ou Reconnaissance de la parole et traduction automatique), et la seconde, située sur Laval (au sein du bâtiment Cérium<sup>2</sup>), elle spécialisée en Ingénierie des EIAH (Environnements Identiques pour l'Apprentissage Humain) et dirigée par Sébastien George. C'est dans cette dernière que j'ai eu l'honneur de réaliser mon stage. *Escape It*, un des projet dirigé par Pierre Laforcade et fait partie des travaux menés par l'équipe EIAH du LIUM sur la conception pédagogique et les jeux sérieux. Le but de ce projet est de donner aux thérapeutes un outil pour faciliter l'apprentissage visuel des enfants atteints d'autisme. Durant ce stage, j'ai pu créer des profils d'enfants et leur gestion, Intégrer de nouvelles scènes de jeu au projet, créer une communication entre le projet Unity et un générateur de niveaux via internet, faire de la retouche d'image et créer des interfaces du jeu pour manipuler ces concepts. Toutes les tâches réalisées pendant la période de stage ont été testées à plusieurs reprises et sont complètement fonctionnelles excepté la scène du salon qui manque peut être un peu de diversité dans les compétences visuelles qu'elle prend en compte.

**5 Mots-clés :**

- **Jeu sérieux**
- **Escape Game**
- **Apprentissage**
- **Autisme**
- **Unity**

## SOMMAIRE

<b>I- Introduction</b>	<b>4</b>
<b>II- Présentation du LIUM</b>	<b>5</b>
II.A- Présentation générale du LIUM	5
II.B- Présentation de l'équipe IEIAH	5
II.C- Cadre de travail	6
II.D- Analyse du LIUM	7
<b>III- Présentation de la mission</b>	<b>7</b>
III.A- Contexte du projet	7
III.B- Sujet du stage	10
III.C- Domaine métier	12
<b>IV- Organisation</b>	<b>13</b>
<b>V- Réalisation de la mission</b>	<b>16</b>
V.A- Gestion des profils	16
V.A.1- Définition d'un profil d'enfant	16
V.A.2- Création et choix de profil	17
V.A.3- Sauvegarde des profils	19
V.A.4- Options	20
V.A.5- Préférences et Ergonomie	22
V.B- Intégration de scènes	24
V.B.1- Création de scènes	24
V.B.2- Intégration de scènes existantes dans un autre projet Unity	25
V.C- Tâches diverses	25
V.C.1- Communication Unity/Générateur	25
V.C.2- Sécurité enfant	28
V.C.3- gestion des fichiers locaux	29
V.C.4- création d'images	30
<b>VI- Ce que le stage m'a apporté</b>	<b>30</b>
<b>VII- Conclusion</b>	<b>32</b>
<b>VIII- Glossaire</b>	<b>33</b>
<b>IX- Références bibliographie/webographie</b>	<b>33</b>

## TABLE DES FIGURES

Figure 1 : Emplacement du Cérium2	7
Figure 2 : anatomie d'une scène	10
Figure 3: Structure du jeu existant	12
Figure 4 : Structure du jeu à réaliser	13
Figure 5 : Schématisation des TSA	15
Figure 6 : capture du trello à deux semaines de travail	16
Figure 7 : capture du trello le 7 Juin 2018	17
Figure 8 : interface utilisateur de création de profil	19
Figure 9 : extrait du code de Authentication.cs	20
Figure 10 : interface utilisateur de choix du profil	20
Figure 11 : Schéma de la sauvegarde et du chargement des profils	22
Figure 12 : Options générales du profil	23
Figure 13 : Options des compétences	24
Figure 14 : Fenêtre de modification des paramètres de la compétence B3	24
Figure 15 : Menu des préférences	25
Figure 16 : Exemple de renforçateur	25
Figure 17 : Paramétrage du prefab Ball	26
Figure 18 : Schéma du processus de génération	28
Figure 19 : script php d'envoi des informations et exécution de la génération	29
Figure 20 : Interface du verrou	31
Figure 21 : Arborescence des fichiers locaux du projet	32

## I- Introduction

Ce stage est un travail réalisé dans le but de valider la deuxième année de mon DUT Informatique à l'IUT de Laval et il a eu lieu au LIUM du 9 Avril 2018 au 22 Juin 2018, soit pendant onze semaines. Le LIUM est un laboratoire informatique dans lequel collaborent des chercheurs titulaires, doctorants et post-doctorants. Au sein du LIUM existe l'équipe Ingénierie des Environnements Informatiques pour l'Apprentissage Humain (IEIAH) qui réalise, entre autres, des travaux sur la conception pédagogique de jeux sérieux.

Les jeux sérieux sont des logiciels avec un but sérieux, comme dans ce cas-ci, la pédagogie, mais qui utilisent des moyens ludiques. Ils permettent donc de rendre plus attrayant un mécanisme d'apprentissage. Les Escape game ou jeux Escape-the-room sont un type de jeu dans lequel le joueur est prisonnier d'une salle et son objectif est d'en sortir. Afin d'atteindre cet objectif le joueur doit en général trouver des éléments cachés, réaliser une série d'action et même résoudre des énigmes, ce pourquoi le genre de l'Escape Game s'adapte plutôt bien aux critères du jeu sérieux.

Les obstacles aux apprentissages des enfants autistes peuvent venir de leur difficultés à généraliser, imiter et déduire l'implicite. Certaines compétences apprises implicitement par des enfants neurotypiques doivent alors être travaillées pour les enfants autistes. L'acquisition, le renfort, et la généralisation de ces compétences sont généralement atteints avec un apprentissage à base d'essais-erreurs. Aujourd'hui, ces enfants ont des prédispositions pour les supports mobiles tactiles, ainsi, afin de faciliter le support de ces apprentissages, l'objectif est de développer un jeu sérieux de type Escape-the-room sur tablette.

Les aspects de recherche de ce projet ont été effectués auparavant et ne constituent pas mon sujet de stage. L'objectif final de ce stage est de pouvoir donner un prototype du jeu sérieux à des thérapeutes pour avoir leurs retours. Il y a un déjà un prototype existant, cependant, il manque de fonctionnalités importantes qui font l'objet de mon stage :

- implémenter la gestion de profils
- gérer la sauvegarde des profils et préférences
- intégrer l'appel au service de génération de scénarios adaptés en ligne
- créer un niveau du jeu
- gérer la progression du joueur dans les différentes compétences
- garder une trace de chaque action effectuée par le joueur
- réaliser les interfaces utilisateur nécessaires
- faire du refactoring et des améliorations sur le code déjà existant

Dans la suite du rapport, vous trouverez en section II une présentation du LIUM, en section III une présentation de la mission, en section IV l'organisation du travail, en section V les travaux réalisés, en section VI ce que le stage m'a apporté, en section VII la conclusion, en section VIII le glossaire, en section IX les références et la bibliographie et enfin en section X vous trouverez les annexes.

## II- Présentation du LIUM

### II.A- Présentation générale du LIUM

Le site du LIUM nous donne des informations sur celui-ci : «Le Laboratoire d'Informatique de l'Université du Mans (LIUM) a été créé il y a environ 30 ans. Il comprend la grande majorité des enseignants-chercheurs en informatique de Le Mans Université et est localisé sur deux sites : le site du Mans dans le bâtiment IC2, et le site de Laval dans les bâtiments de l'IUT. Depuis sa création, les thèmes de recherche majeurs du LIUM sont l'Ingénierie des Environnements Informatiques pour l'Apprentissage Humain (IEIAH) et le traitement automatique de la parole (LST).»

(<https://lium.univ-lemans.fr/lium/le-laboratoire/>)

Le LIUM est composé de 60 personnes dont 4 personnels administratif et techniques, 28 enseignants chercheurs et 26 doctorants et post-doctorants. Il est en collaboration avec d'autres laboratoires, qu'ils soient français, européens, ou internationaux. De plus, le LIUM possède de nombreux partenariats industriels avec notamment Airbus, Orange Lab ou encore Voxolab. Il est aussi notable que le LIUM fait partie du département numérique de l'Université Bretagne Loire, qu'il est membre de l'Altlanstic 2020 (qui fédère l'écosystème économique du numérique dans les Pays de Loire), ainsi que de l'institut d'informatique Claude Chappe.

### II.B- Présentation de l'équipe IEIAH

Ayant intégré l'équipe du projet *Escape It* pendant mon stage, j'ai également intégré l'équipe IEIAH décrite ainsi sur le site du LIUM :

« Les travaux de l'équipe IEIAH, dirigée par Sébastien GEORGE, s'inscrivent principalement dans l'Apprentissage Humain (EIAH) mais aussi dans les domaines du Génie Logiciel et des Interactions Humains Machines. Plus précisément, l'équipe est spécialisée sur l'ingénierie des EIAH dont l'originalité réside dans une vision du processus d'ingénierie comme relevant de la conception continue (i.e. se poursuivant dans l'usage).

Le projet scientifique fondamental de l'équipe est d'élaborer un socle scientifique pour le développement et l'ingénierie des systèmes informatiques pour l'enseignement et l'apprentissage. Cet objectif structure la totalité des actions de l'équipe, tant en termes de recherche que d'animation de la recherche. »

(<https://lium.univ-lemans.fr/lium/ieiah/>)

L'objectif principal de cette équipe est d'intégrer les enseignants et les formateurs (non chercheurs) dans le processus de conception d'un EIAH, et leurs travaux sont orientés sur l'analyse de la situation d'apprentissage, guidée par les besoins d'observation des enseignants. l'équipe propose des solutions pour permettre la description d'indicateurs pédagogiques et de données d'observation mais également pour collecter les traces d'usage. L'équipe s'intéresse également à l'utilisation de tables

interactives, réalité mixte et interfaces tangibles notamment dans le contexte des Serious Games.

## II.C- Cadre de travail

Le Cérium2 est une plateforme du LIUM située sur LAVAL occupée par une partie de l'équipe IEIAH d'une superficie de 146m<sup>2</sup> et dispose d'une salle d'expérimentation. C'est dans ces locaux que j'ai pu travailler. Dans le bureau qui m'a été attribué, il y a six places, je vivais donc au quotidien avec cinq autres personnes dont une qui travaillait aussi dans le cadre du projet *Escape It* : Youness Laghouaouta. Mon autre collègue n'était autre que mon encadrant Pierre Laforcade. J'ai davantage travaillé avec M. Laforcade qu'avec M. Laghouaouta qui lui travaillait sur des aspects de recherche (un générateur de scénarios en ligne) du projet non directement intégrés dans le prototype que je devais améliorer. Afin de travailler ensemble, il était nécessaire de faire des réunions. Généralement, moi et M. Laforcade organisons des réunions entre nous deux pour faire l'état du travail effectué et des objectifs à atteindre pour la prochaine réunion, cependant il y a eu des exceptions notamment lorsqu'il fallait faire la liaison entre le jeu et le générateur ce qui a nécessité la présence de M. Laghouaouta. Autrement, chaque mois, a lieu une réunion d'équipe qui concerne tous les membres de l'équipe IEIAH au sujet de l'état administratif de la structure principalement qui s'ensuit généralement de la présentation des avancements d'un des membres dans son projet (j'ai pu assister à deux de ces réunions).



Figure 1 : Emplacement du Cérium2



## II.D- Analyse du LIUM<sup>1</sup>

Le LIUM est un laboratoire de recherche en informatique affilié à l'Université du Maine, qui est un EPSCP (établissement public à caractère scientifique, culturel et professionnel). Ainsi, le laboratoire ne possède pas de capital propre, mais les fonds qu'il perçoit par l'état sont alloués par projet.

<b>Forces</b>	<b>Faiblesses</b>	<b>Opportunités</b>	<b>Menaces</b>
Une large équipe comprenant 28 enseignants-chercheurs  Une expertise dans le domaine de l'informatique  Des collaborations plus fréquentes avec des chercheurs étrangers	La quasi-totalité des membres ont des services d'enseignement et des responsabilités administratives importantes	créer des relations avec des chercheurs étrangers	Coût de plus en plus important de la recherche  Difficulté à trouver des financements extérieurs

Le LIUM adopte une stratégie générale de spécialisation dans le domaine des EIAH.

Le LIUM fait des appels à projets pour distribuer ses ressources humaines dans les différents projets. Il lui arrive de recruter des stagiaires et des doctorants et post-doctorants. Chaque équipe est autonome, organise ses réunions et gère son budget seule.

---

<sup>1</sup> Réalisé en collaboration avec Guillaume Ciret et Fabien Lecrubier

### III- Présentation de la mission

#### III.A- Contexte du projet

*Escape It* un projet dirigé par Pierre Laforcade et fait partie des travaux menés par l'équipe IEIAH du LIUM, en collaboration avec l'association Cocci'Bleue, sur la conception pédagogique et les jeux sérieux. Le but de ce projet est de donner aux thérapeutes un outil pour faciliter l'apprentissage visuel des enfants atteints d'autisme. Les compétences visuelles concernées par ce projet sont celles identifiées comme :

- B3 : Appariement des objets à des objets identiques
- B4 : Appariement des objets à des images
- B8 : Classer des items non identiques
- B9 : Placer des pièces sur un modèle
- B13 : Reproduire une séquence visuelle en présence du modèle
- B19 : Trier par catégorie
- B25 : Sériation

En effet le site du projet *Escape It* le décrit comme tel : «Le jeu sérieux d'apprentissage se présente comme un jeu mobile pour grandes tablettes (10") Android/iOS. Le jeu est en 2D et propose des graphismes de type "cartoon". Le gameplay actuel utilise le glisser-déposer (drag'n drop). Le jeu a été développé avec le moteur Unity. Il permet à l'enfant, accompagné d'un thérapeute/parent, de jouer une partie de 3 à 5 niveaux. Chaque niveau mettra se présentera sous la forme d'un lieu du quotidien dont la sortie est fermée. Pour ouvrir la porte et passer au niveau suivant, l'enfant devra trouver des "objets" et les placer aux bons endroits. Chaque niveau/scène de jeu vise un seul objectif d'apprentissage de type performance visuelle (sous-ensemble ré-adapté des performances visuelles décrites dans le guide ABLLS-R)» ([https://projets-lium.univ-lemans.fr/Escape It/](https://projets-lium.univ-lemans.fr/Escape%20It/))

Le thème de ce jeu sérieux est la maison. Chaque niveau est une salle de la maison (chambre, cuisine, salon, etc..) dont le but est d'en sortir. Afin de finir le niveau, le joueur doit replacer les bons objets aux bons endroits. Les niveaux sont créés par un générateur en fonction du profil du joueur, de la façon dont les compétences à travailler peuvent se traduire sur la manière de finir le niveau et des compétences que peut traiter chaque niveau.

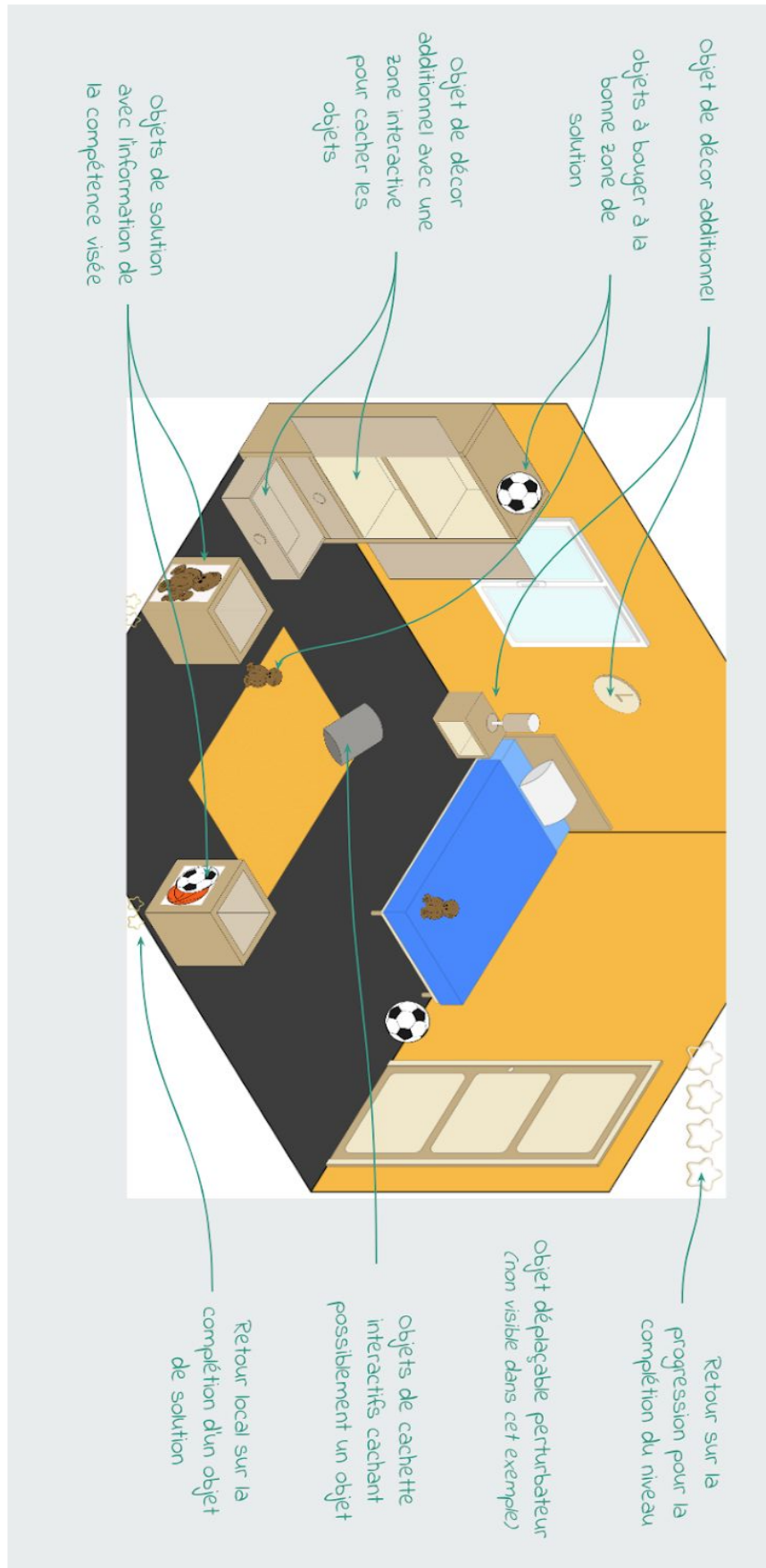


Figure 2 : anatomie d'une scène (présentation "CSEDU'18 - ESCAPE IT!")

Le site du LIUM nous éclaire sur le sujet les recherches liés à *Escape It* : «Ce projet aborde la complexité de la conception de jeux sérieux pédagogiques lorsqu'ils sont adaptés à un public et à un domaine d'apprentissage spécifiques, et lorsque le besoin d'adaptation (dynamique) des jeux au profil de l'apprenant est nécessaire. Les verrous scientifiques portent sur (1) la formalisation des éléments statiques et dynamiques du jeu selon différentes dimensions inter-reliées (domaine d'apprentissage, approches pédagogiques, mécaniques de jeu), et (2) l'exploitation de ces modèles exécutables à des fins de validation avec les experts métiers impliqués dans des approches itératives de co-conception des jeux sérieux. L'objectif est alors d'étudier et de proposer des solutions IDM (ingénierie dirigée par les modèles) afin de faciliter la formalisation et l'exploitation des modèles produits afin de supporter les besoins en adaptation (génération de scénarios de jeux adaptatifs) et en validation (prototypes dirigés par les modèles prenant en compte la sémantique opérationnelle du jeu). Le domaine d'application du projet consiste à concevoir et développer un jeu sérieux à destination d'enfants autistes.»  
([https://lium.univ-lemans.fr/Escape It/](https://lium.univ-lemans.fr/EscapeIt/))

Son contexte sociétal est décrit sur le site du projet : «Le projet s'inscrit dans un cadre sociétal d'intérêt public en participant à proposer des outils-soutiens pour l'apprentissage des enfants autistes.

Les derniers plans autisme, les recommandations de la Haute Autorité de Santé (HAS), et surtout les avancées de prises en charge dans d'autres pays, ont mis en avant l'importance de l'apprentissage comme meilleur « soin » possible. Il est dorénavant reconnu qu'il faut dés-institutionnaliser les enfants autistes et permettre au plus vite leur inclusion scolaire. Pour cela l'effort doit se porter sur un diagnostic le plus précoce possible (avant 18 mois) afin de pouvoir au plus tôt proposer des prises en charges comportementales et cognitives visant l'apprentissage de compétences élémentaires nécessaires à l'inclusion scolaire.

C'est dans ce contexte que le projet se situe. Il vise à supporter la généralisation des apprentissages des compétences visuelles (appariement, tri, catégorisation, sériation, etc.) en proposant un jeu sérieux qui pourra être initié en séance avec le thérapeute et continué avec une supervision parentale à la maison.

Cette approche permet de réduire les coûts pour les familles et réduire le temps d'apprentissage pour l'enfant lorsque le travail est réalisable avec des professionnels formés aux bonnes pratiques recommandées par l'HAS, non remboursés actuellement malheureusement.»

([https://projets-lium.univ-lemans.fr/Escape It/vue-generale/](https://projets-lium.univ-lemans.fr/EscapeIt/vue-generale/))

### III.B- Sujet du stage

Quand je suis arrivé sur le projet *Escape It*, il existait déjà un prototype du jeu développé avec Unity en C# mais ne concernant que quatre compétences visuelles et deux scènes jouables (la cuisine et la chambre). Le générateur de niveaux existait mais les scénarios de jeux étaient pré-générés, et le jeu ne communiquait pas avec le générateur. Le concept de profil n'existait pas dans le jeu, les options non plus, de même pour le système de progression dans l'apprentissage, et, par conséquent, les Interfaces Utilisateur correspondantes n'existaient pas non plus. Enfin, à ce moment là, *Escape It* avait quelques problèmes notamment liés à la résolution des tablettes et à l'affichage du jeu.

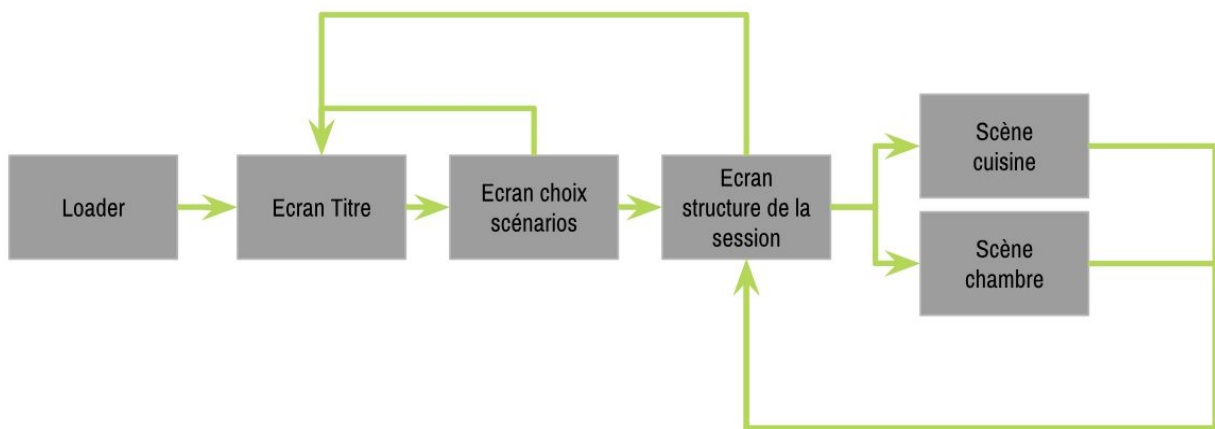


Figure 3: Structure du jeu existant (présentation “Projet Escape It! - Evolution prototypes”)

Sur la figure 3 ci-dessus figure un schéma de navigation qui donne une idée précise de la structure et du contenu de l'existant. En premier lieu nous avons *LoaderGameManager* (Loader sur la figure 3) qui permet de créer une scène de jeu à partir d'un fichier xml, il est appelé en premier et reste en arrière-plan après avoir appelé l'écran titre, et sera rappelé quand le jeu aura besoin de générer une scène. Depuis l'écran titre, le prototype permettait d'atteindre une fenêtre où il proposait différents scénarios pré-générés (un pour chaque compétence et pour chaque niveau de maîtrise). Une fois le scénario choisit, le prototype affichait une interface où l'utilisateur pouvait voir où il en était dans le scénario et pouvait lancer le prochain niveau. En lançant un nouveau niveau, le *LoaderGameManager* est appelé et génère une scène de jeu Unity en fonction de sa lecture du fichier de scénario pré-généré.

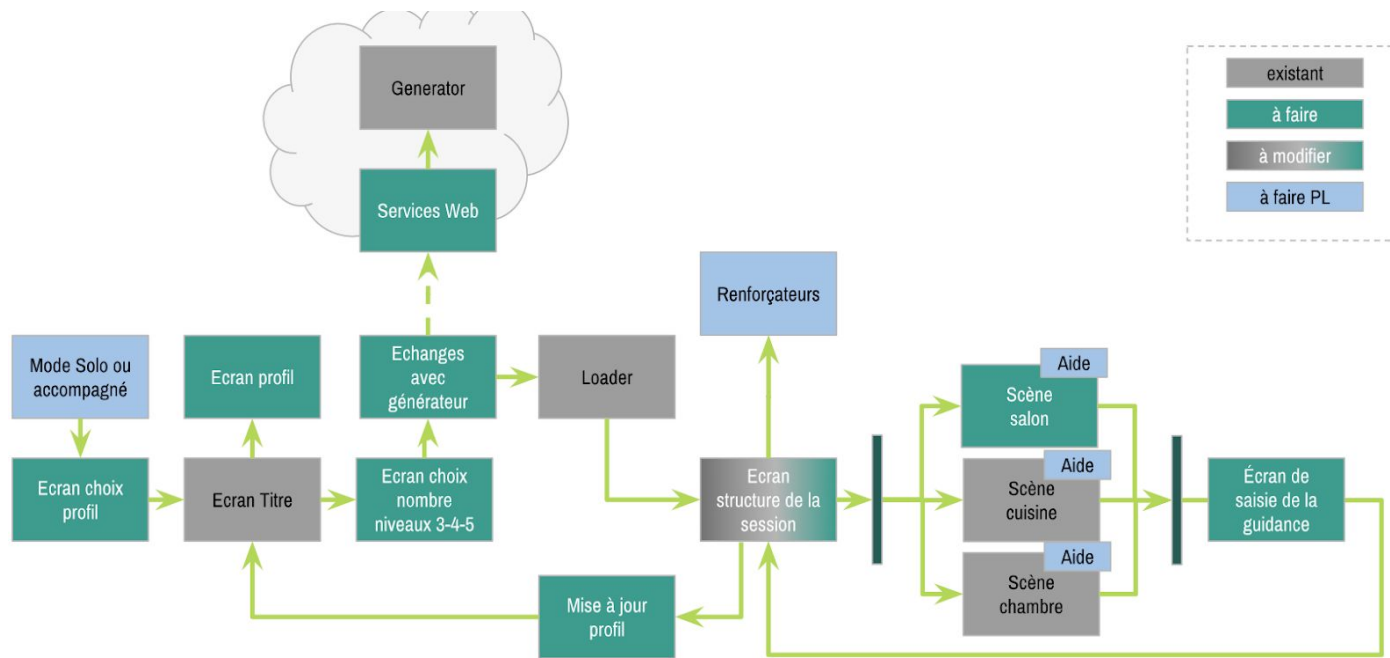


Figure 4 : Structure du jeu à réaliser (présentation “Projet Escape It! - Evolution prototypes”)

Les aspects de recherche de ce projet ont été effectués auparavant et ne constituent pas mon sujet de stage. L’objectif final de ce stage est de pouvoir donner un prototype du jeu sérieux à des thérapeutes pour avoir leurs retours. Pour le moment, le projet *Escape It* manque de fonctionnalités cruciales :

- Gérer les profils (création, modification, suppression, utilisation)
- gérer la sauvegarde des profils et préférences
- intégrer l’appel au service de génération de scénarios adaptés en ligne
- créer une scène jouable (sur le thème du salon)
- gérer la progression du joueur dans les différentes compétences
- réaliser les interfaces utilisateur nécessaires
- faire du refactoring et des améliorations sur le code déjà existant

La gestion des profils est une tâche décomposable qui va demander la création de plusieurs interfaces utilisateur comme le montre la figure 4 ci-dessus. En effet, on peut déjà remarquer la présence d’un écran pour le choix du profil et un autre pour le paramétrage des profils (écran profil sur la figure 4). On peut ensuite remarquer la présence d’une grande carence dans la structure du jeu, celle de l’appel au service de génération de scénarios adaptés en ligne. Ceci va demander la création d’écrans pour saisir le choix de l’utilisateur concernant la longueur du scénario, un système de création et d’enregistrement de fichier, ainsi que de communication avec un service web. Enfin nous ajouterons des fonctionnalités dans l’écran de la structure de la session, ainsi qu’un autre système pour mettre à jour le profil de l’enfant suivant ses succès dans la complétion des niveaux.

Pour réaliser ces tâches, j'ai eu besoin d'utiliser plusieurs logiciels. Celui que j'ai utilisé le plus est Unity. Unity est un logiciel de développement de jeux-vidéo multi-plateforme 2D et 3D et a, pour langage de programmation principal le C#. Pour créer les interfaces utilisateur, il a parfois fallu créer des images pour des boutons ou autre, et pour ce, j'ai utilisé le logiciel GIMP qui est un logiciel d'édition et de retouche d'image. Enfin, pour la création d'éléments de décor des scènes, j'ai utilisé le logiciel Affinity Designer qui est un éditeur graphique vectoriel.



**Unity**



**Affinity Designer**



**gimp**

### **III.C- Domaine métier**

Pour comprendre le contexte de travail, il est important de comprendre certains termes, ce qui facilitera la compréhension des tâches à réaliser et des contraintes de travail. Il va tout d'abord être nécessaire de connaître la définition de l'autisme, ici décrite par le site de Larousse : «Trouble du développement neurologique caractérisé par une altération des interactions sociales (repli pathologique sur soi), de la communication (langage) et du comportement.»

<https://www.larousse.fr/dictionnaires/francais/autisme/6565>

Pour observer les particularités de l'autisme les enfants autistes et leurs comportements sont comparés à ceux des enfants dits neuro-typiques qui sont l'opposé même des enfants présentant des troubles autistiques et sont considérés comme norme. Il est d'ailleurs important de noter qu'il existe tout un spectre de troubles autistiques et non un seul.

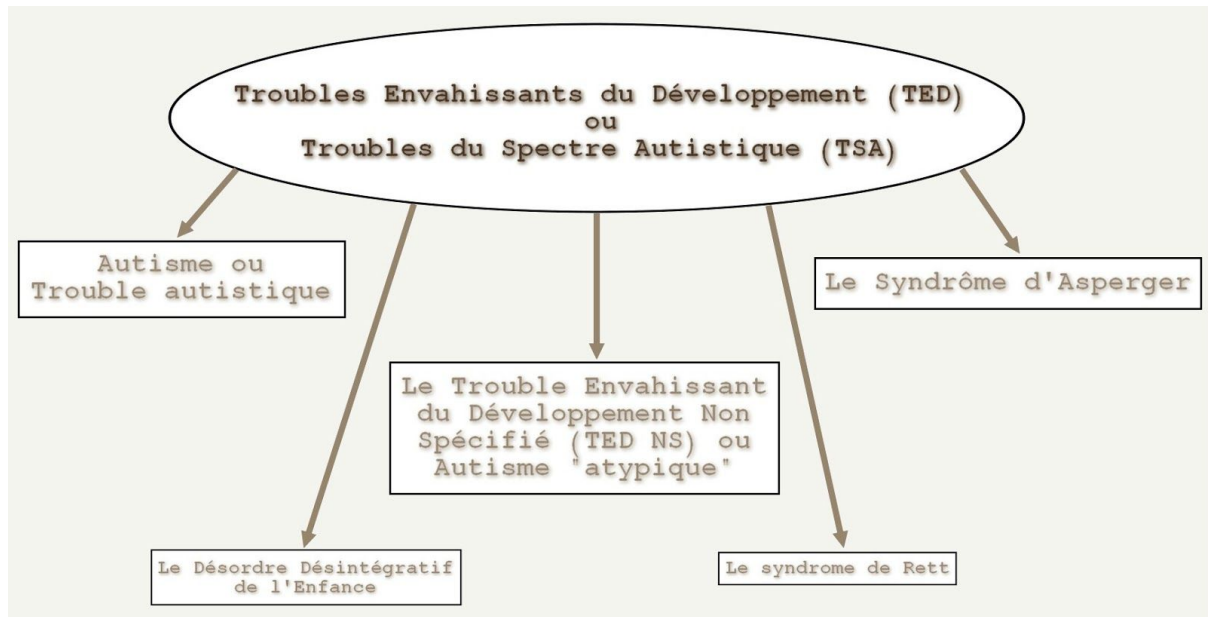


Figure 5 : Schématisation des TSA

(<http://iletedunefois.canalblog.com/archives/2013/07/06/27579361.html>)

## IV- Organisation

Si nous sommes plusieurs à travailler sur le projet, il n'y a pourtant pas un aspect où plusieurs personnes ont travaillé simultanément, ce qui signifie que le travail que j'ai réalisé a été fait seul, avec les retours et directives de mon encadrant évidemment. Cependant, lorsque, dans le projet il est question de mettre en lien les travaux de plusieurs personnes, il devient nécessaire d'effectuer des réunions et de prévoir des moments de travail en groupe.

Ayant défini un nombre précis et quantifiable de tâches à réaliser pendant mon stage avec Pierre Laforcade, il était assez simple de planifier des rendez-vous réguliers afin de constater l'avancement des tâches à réaliser de me donner des directives supplémentaires si le résultat n'était pas celui attendu. Ces rendez-vous s'effectuaient au moins une fois par semaine. Afin d'avoir une vision globale de l'avancement du travail à faire, nous utilisons un trello, tenu à jour régulièrement. Cependant, même avec cette vision globale des tâches à réaliser et de l'objectif à atteindre avant la fin du stage et notre organisation hebdomadaire, la date de fin du stage (à savoir le 22 Juin 2018) n'était pas notre seule date limite. En effet, le 7 Juin 2018 a eu lieu la visite du personnel de l'IUT de LAVAL au Cérium2 pour découvrir le monde de la recherche, date pour laquelle nous nous étions donné l'objectif d'avoir un prototype jouable, fonctionnel avec le moins de bogues possible.



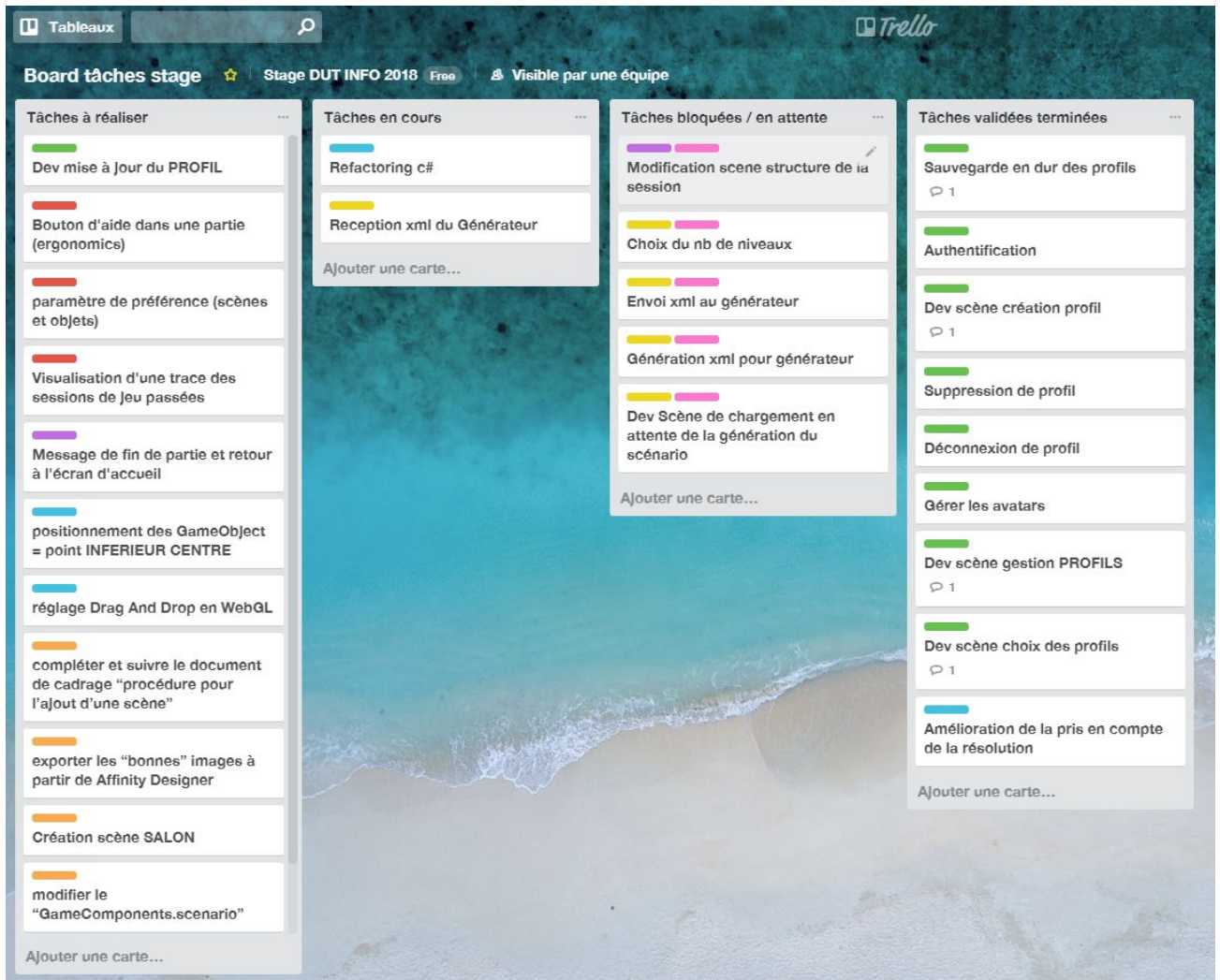


Figure 6 : capture du trello à deux semaines de travail

L'organisation au sein du Trello est basique. En effet, les tâches peuvent être rangées dans quatre catégories :

- Tâches à réaliser
- Tâches en cours
- Tâches bloquées/en attente
- Tâches validées terminées

Les différentes tâches se voient affiliée à une ou plusieurs couleurs (sauf exception). La première couleur sert à indiquer l'objectif dans lequel est comprise la tâche (par exemple, la tâche "Gérer les avatars" possède la couleur verte qui correspond à "gestion des profils"). La seconde couleur sert à connaître le statut de la tâche si elle est dans la catégorie des "tâches bloquées/en attente" (pour savoir si elle est bloquée car elle a besoin de la complétion d'une autre tâche ou si elle est simplement en attente de validation).

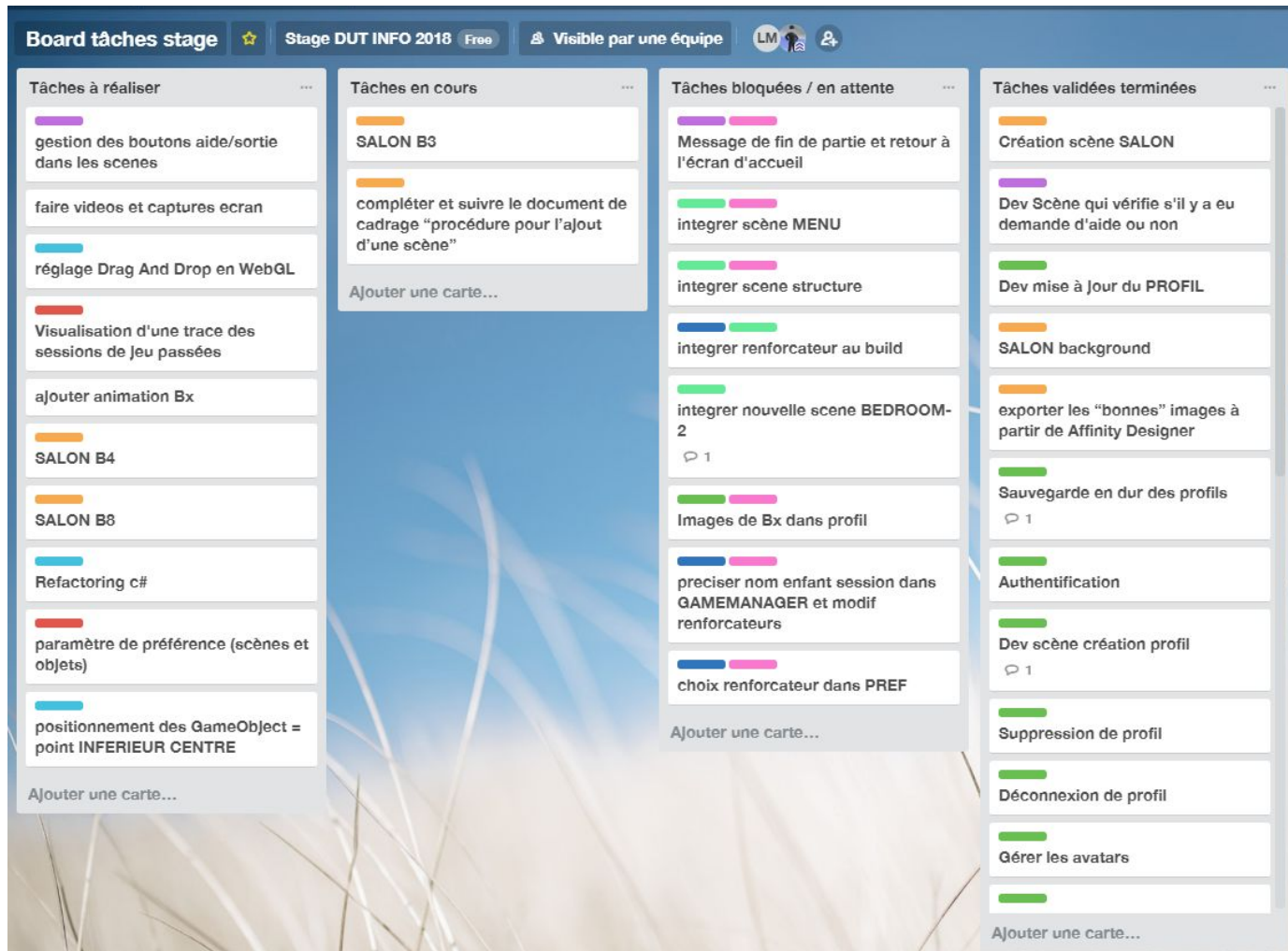


Figure 7 : capture du trello le 7 Juin 2018

En plus du Trello pour la mise en commun de l'avancée des différentes tâches, un Google Drive partagé a aussi été mis en place contenant des ressources mises à disposition. Parmi ces ressources, figurent des diapositives de présentation du projet, de la documentation sur le projet et sur Affinity Designer, Unity, etc, et enfin, des documents qui permettent de rendre compte de l'avancement du projet comme le document qui répertorie les différentes compétences visuelles gérées par les différentes scènes de jeu (chambre, cuisine, salon) et à quels niveaux de difficulté elles sont gérées. De plus, le ce Google Drive me permet d'héberger ce rapport afin d'optimiser la capacité de mon encadrant à faire des remarques dessus. Il est à noter qu'il m'est arrivé de réaliser des vidéos et des captures d'écran afin de montrer et de rendre compte de l'avancement des tâches pour avoir des retours plus rapidement.

## V- Réalisation de la mission

### V.A- Gestion des profils

#### V.A.1- Définition d'un profil d'enfant

En arrivant sur *Escape It*, le concept de profil n'existait pas cependant, il est indispensable dans le sens où le but d'*Escape It* est de générer dynamiquement des scénarios d'apprentissage adapté à l'enfant en fonction de sa maîtrise des différentes compétences visuelles proposées par le jeu. J'ai donc créé ce concept de profil, de façon à ce qu'il contienne les informations nécessaires à la création des scénarios de jeu par le générateur (excepté une que nous verrons dans la partie V.C.1), mais aussi le minimum d'informations indispensables à un profil.

Chaque profil possède donc :

- Un identifiant
- Le nom de l'enfant à qui appartient le profil
- Un avatar
- Sept compétences visuelles
- La date et l'heure de la dernière utilisation

L'identifiant servira à déterminer quel fichier de préférences et quels fichiers de renforçateur correspond à ce profil, ce que nous détaillerons dans la partie V.A.4. Il est à noter que l'enfant ne peut pas choisir son avatar depuis les fichiers de sa tablette mais qu'il doit faire un choix dans une liste de seize avatars prédéfinie. Si chaque profil possède sept compétences visuelles c'est parce que le jeu ne fera jamais travailler d'autres compétences que les compétences citées en III.A, et car la présence de chacune de ces compétences dans le profil est nécessitée par le générateur (comme nous le verrons en V.C.1).

Chaque compétence contient :

- Le statut de son activité (active ou non)
- Le degré de maîtrise de l'enfant dans la compétence en question
- Le statut de l'acquisition de la compétence par l'enfant (acquis ou en cours d'acquisition)
- Le nombre de succès consécutifs de l'enfant dans les épreuves concernant la compétence en question au degré de maîtrise actuel de celle-ci.

Il existe cinq degrés de maîtrise dans une compétence :

- Débutant
- Élémentaire
- Intermédiaire
- Avancé
- Expert

Le nombre de succès consécutifs permet de faire passer le degré de maîtrise d'une compétence par l'enfant au degré suivant, augmentant progressivement la

difficulté des scénarios joués par l'enfant (comme nous le verrons dans la section V.C.1). Il est également à noter qu'il n'y a pas de notion d'échec dans ce jeu, pour éviter toute frustration pour l'enfant. Le nombre de succès consécutifs représente en fait le nombre de fois où l'enfant a été capable de résoudre l'épreuve seule, sans l'aide du thérapeute (nous partons du principe que toutes les utilisations d'*Escape It* par un enfant sont supervisées par un thérapeute). Le thérapeute a une interface qui lui est exclusive afin de donner l'information concernant la présence d'aide ou non de sa part, dans la résolution de chaque niveau, au jeu. Cela lui permettra, à nouveau, d'éviter une quelconque frustration chez l'enfant. Nous aurons l'occasion d'aborder cette partie en détail dans la section V.C.2. Enfin, le statut de l'acquisition de la compétence et son degré de maîtrise permet également au générateur de décider quelles compétences il est judicieux d'incorporer au prochain scénario, et nous le verrons également plus en détail dans la partie V.C.1.

### V.A.2- Création et choix de profil

Une fois que le concept de profil existe, il nous faut un système pour pouvoir l'exploiter, mais, avant tout, un système pour déterminer quel profil doit être utilisé. Pour cela, j'ai créé un moyen de créer et un moyen de choisir un profil. Pour la création d'un profil, il était nécessaire d'avoir le moins d'actions possibles à effectuer pour atteindre notre but. J'ai donc décidé de mettre, lors de la création d'un profil, toutes ses compétences visuelles à une configuration par défaut (à savoir active, non acquise et au niveau débutant), son horodatage à celui de sa création, son avatar au premier avatar de la liste (celui par défaut donc), pour que seul le nom de l'enfant soit à saisir dans une boîte de dialogue dans l'interface utilisateur liée à la création d'un profil.

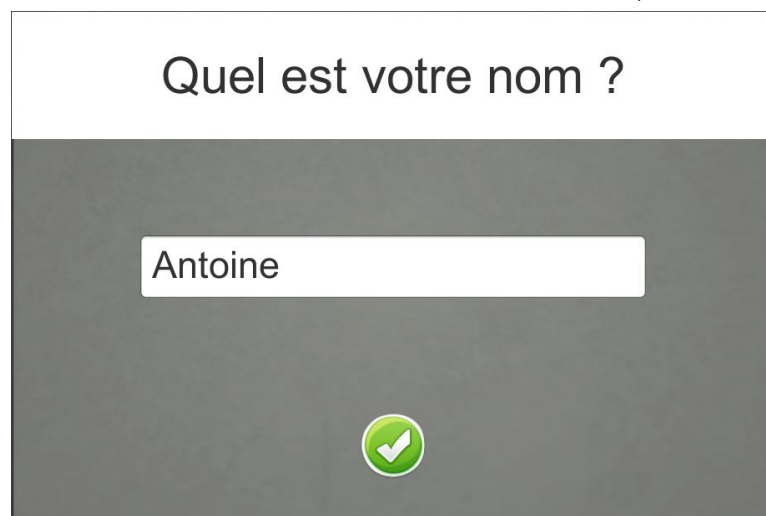


Figure 8 : interface utilisateur de création de profil

Pour le choix du profil, il m'était aussi nécessaire de créer une interface utilisateur, mais celle-ci avait également besoin d'un moyen de répertorier tous les profils existants au sein du jeu mais aussi de déterminer quel est le profil utilisé tout au long de la session après le choix. Pour cela, à chaque création de profil, ce profil est ajouté à une liste des profils existants. à chaque modification de cette liste ou des profils dans cette liste, celle-ci est réordonnée en mettant en premier les profils dont le dernier horodatage est

le plus récent. Pour prendre en compte le choix du profil, le profil correspondant dans la liste de tous les profils créés dans le jeu est affecté à la classe *Authentication* via sa fonction *Connect()*, et accessible en lecture partout ailleurs dans le code, à l'aide de sa fonction statique *GetCurrentProfile()*.

```
public class Authentication {  
  
    private static Profile currentProfile;  
  
    public static void Connect(Profile p)  
    {  
        currentProfile = p;  
    }  
    public static void QuitCurrentProfile()  
    {  
        currentProfile = null;  
    }  
    public static Profile GetCurrentProfile()  
    {  
        return currentProfile;  
    }  
}
```

Figure 9 : extrait du code de Authentication.cs

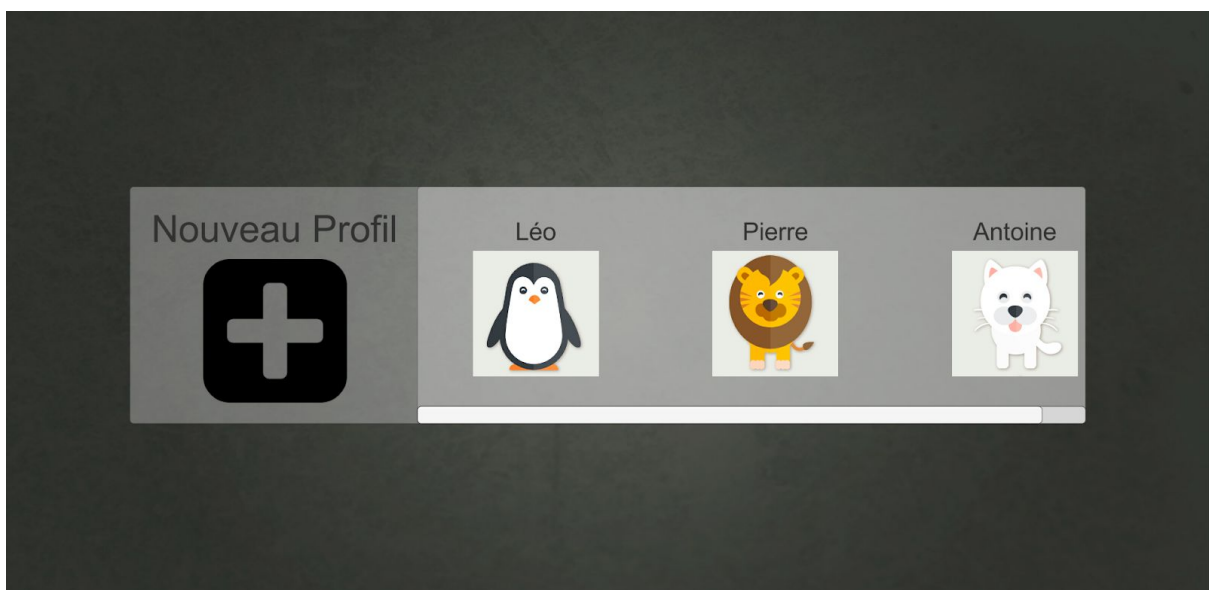


Figure 10 : interface utilisateur de choix du profil

L'interface utilisateur du choix du profil dont nous avons parlé ci-dessus (et dont un aperçu, un peu inspiré de Netflix, se trouve en figure 10) est décomposée en deux parties. La première, à droite est un bouton qui sert d'appel à la boîte de dialogue (figure 8) qui permet la création d'un nouveau profil. La deuxième permet d'afficher des boutons sur lesquels figurent les noms et avatars de tous les profils répertoriés par le jeu. Quand il y a plus de trois profils répertoriés, l'interface ne les affiche pas tous à la fois mais propose une barre de défilement en bas qui permet d'accéder à n'importe quel profil sans avoir à tous les afficher en même temps. Les profils s'affichent de gauche à droite dans l'ordre de la liste dans laquelle ils sont répertoriés, c'est-à-dire celui utilisé le plus récemment est placé le plus à gauche. Nous supposons que c'est le thérapeute qui possède la tablette sur laquelle le jeu est installé, et qu'il ne voit pas plus d'une vingtaine d'enfants (ce qui est déjà beaucoup pour un seul thérapeute), ce qui justifie l'absence de nécessité d'une barre de recherche pour faire de la recherche par nom, et d'autres éléments de distinction entre les différents profils.

### V.A.3- Sauvegarde des profils

La sauvegarde des profils ne s'effectue pas via une base de données et ce pour deux raisons. La première est le fait que la quantité d'informations à stocker est extrêmement faible, de plus, faire un lien entre Unity et une base de données (qu'il faut par ailleurs administrer, concevoir, créer, sans même parler des droits liés à cet base de données -déclaration à la CNIL, etc.) peut s'avérer long et fastidieux. La seconde est que nous n'avons pas besoin d'un accès à distance pour les profils de chaque enfant, si l'on considère que l'enfant ne voit pas une dizaine de thérapeutes différents. La sauvegarde par base de données se fera peut-être dans des versions ultérieures cependant elle ne semble pas être un bon choix en ce qui concerne le prototype que nous avons pour but d'avoir réalisé à la fin de ce stage.

La sauvegarde ici consiste donc en la création d'un fichier xml en local (nous verrons la sauvegarde des fichiers locaux du projet Unity dans la section V.C.3). La sauvegarde de tous les profils répertoriés se fait dans un seul et même fichier xml (celui-ci est créé lors de la première sauvegarde et écrasé les sauvegardes suivantes). Pour transcrire toutes ces données en xml, j'ai utilisé l'outil fourni par Unity : le *XMLSerializer*. Ce dernier fera notre travail quant aux balises et à la structure du fichier xml, il ne nous manque plus que lui donner des instances sérialisables (il est possible de modifier les balises qu'il choisi et de lui en imposer, ce que nous verrons dans la partie V.C.1). *XMLSerializer* sérialisera uniquement les attributs avec une portée publique de la classe sérialisable, ce pourquoi il ne serait pas correct de rendre directement les profils sérialisables. Nous allons donc plutôt créer des classes similaires mais sérialisables, et instanciables avec nos instances déjà présentes de profils (nous rendrons d'ailleurs les profils instanciables par les profils sérialisables pour la lecture du fichier de sauvegarde). Ainsi à chaque sauvegarde, pour tous les profils répertoriés, on instancie leur version sérialisable et on la met dans une liste de profils sérialisables, elle-même sérialisable, que l'on donne au *XMLSerializer*, qui créé le fichier xml.

Dans le sens inverse, on donne le fichier de sauvegarde au *XMLSerializer* qui va le désérialiser et nous rendre la liste sérialisable de profils sérialisables que nous lui avons donné. ainsi il suffit d'instancier les profils avec les profils sérialisables et de les mettre dans la liste des profils répertoriés. On peut d'ailleurs retrouver ce fonctionnement schématisé en figure 11, avec en bleu la sauvegarde et en rouge le chargement des profils.

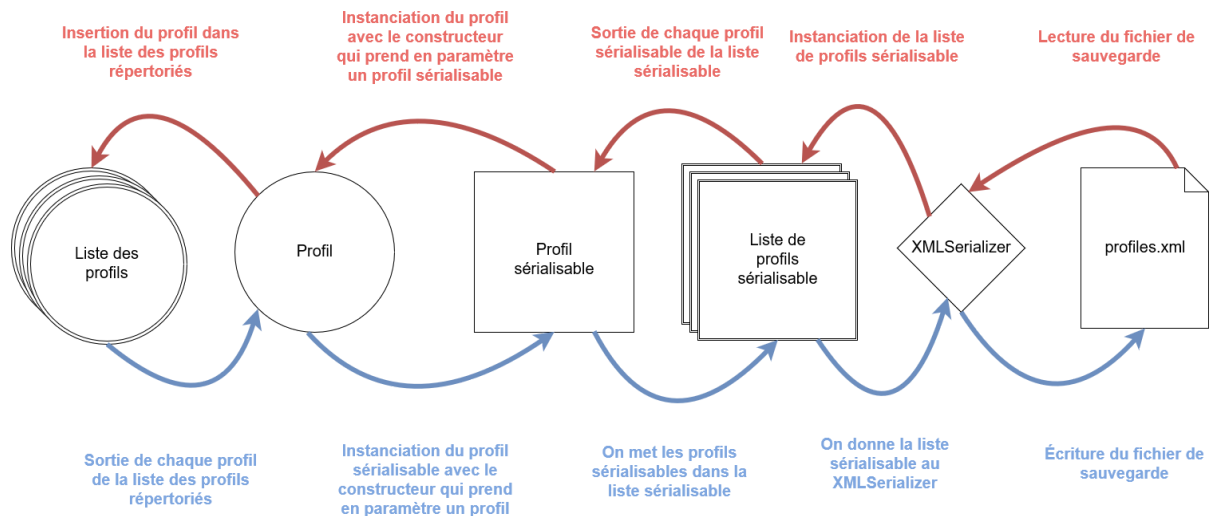


Figure 11 : Schéma de la sauvegarde et du chargement des profils

#### V.A.4- Options

Comme vu précédemment en section V.A.1, lors de la création des profils, seul le nom est paramétrable, pourtant, il serait arrangeant de pouvoir changer le niveau de maîtrise de l'enfant, s'il n'a plus un niveau débutant dans certaines compétences visuelle, et , de plus, si le thérapeute ne veut pas travailler telle ou telle compétence visuelle, il devrait être à même de pouvoir les désactiver. Enfin, il peut parfois s'avérer utile de modifier le nom de l'enfant (coquille, homonyme) et l'enfant peut avoir envie d'un nouvel avatar. Ce pourquoi j'ai créé le menu des options dans lequel tout ceci est modifiable à volonté.

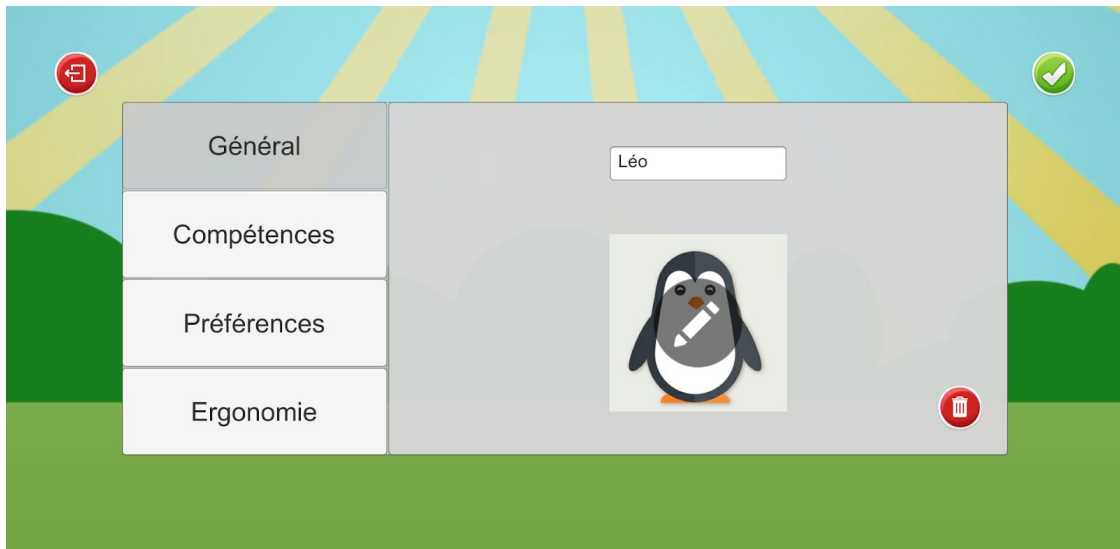


Figure 12 : Options générales du profil

Dans cette partie du menu des options il est possible d'éditer le nom de l'enfant, et, en appuyant sur le bouton d'édition de l'avatar, il est possible d'ouvrir une fenêtre qui permet de modifier l'avatar du profil. Il est également possible d'y supprimer le profil. Depuis toutes les fenêtres des options il est possible d'atteindre les autres via le menu latéral, de se déconnecter du profil, ou de valider les modifications et sortir des options pour rejoindre le menu principal .



Figure 13 : Options des compétences



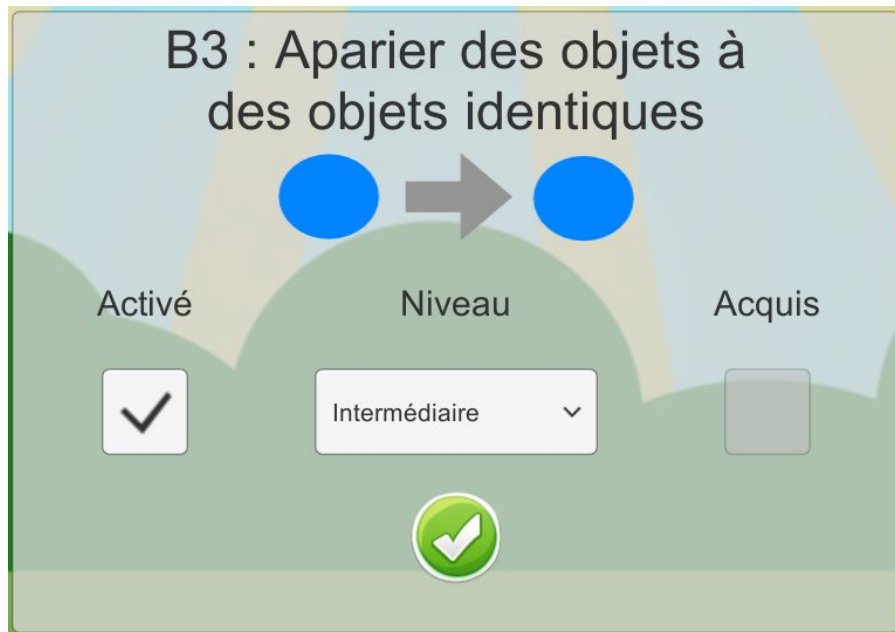


Figure 14 : Fenêtre de modification des paramètres de la compétence B3

Dans les options des compétences, nous avons un aperçu de l'état de chaque compétence avec une liste. Chaque élément de la liste possède un bouton de modification qui permet d'ouvrir une fenêtre de saisie des paramètres de la compétence (figure 14), et le statut de ses caractéristiques :

- Le nom de la compétence
- La description de la compétence
- l'activité ou non de la compétence
- Le niveau de maîtrise de la compétence
- L'acquisition ou non de la compétence
- Le nombre de succès consécutifs

L'activité, le niveau de maîtrise et l'acquisition de la compétence sont représentés par l'absence ou le contenu de l'élément coloré visible dans la figure 13, tandis que le nombre de succès consécutifs est indiqué par le nombre d'étoiles. Pour que ce menu affiche correctement les informations du profil, on utilise celui affecté à Authentication.

Afin d'enregistrer les modifications, on désaffecte le profil de l'enfant à Authentication, on récupère les informations entrées par l'utilisateur et on les utilise pour modifier le premier profil de la liste des profils répertoriés (celui que l'on vient de désaffecter) et on effectue la sauvegarde et le chargement des profils (vue en section V.A.3). Ensuite, on réaffecte le premier profil de la liste des profils répertoriés à Authentication et on met à jour les informations affichées dans les options.

#### V.A.5- Préférences et Ergonomie

Le menu des préférences (figure 15) permet lui de modifier le thème musical du jeu, le thème des renforçateurs (les renforçateurs, figure 16, sont des scènes supplémentaires dans lesquels l'enfant peut débloquer et collectionner des images

lorsqu'il finit un scénario de jeu, ce qui a pour but de le motiver à travailler ses compétences visuelles). Le menu des préférences permet également de modifier un code que l'on verra plus en détail en section V.C.2.

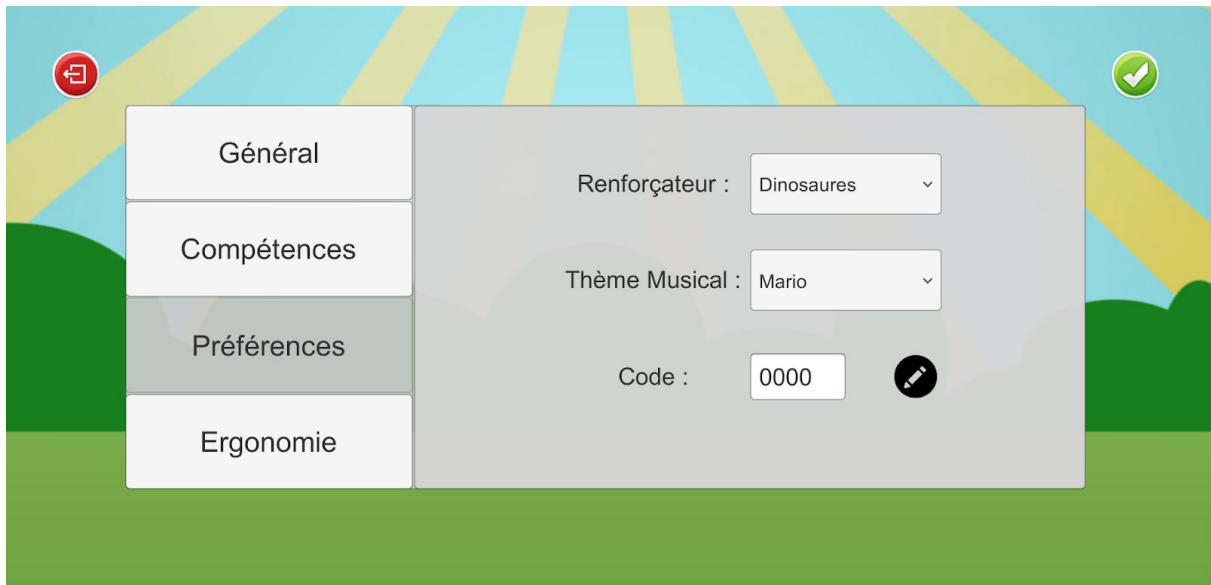


Figure 15 : Menu des préférences



Figure 16 : Exemple de renforceur

Enfin, le menu d'Ergonomie existe, mais seulement en prévention de futures fonctionnalités qui ne rentreront pas dans les catégories déjà existantes. Par exemple, si l'enfant ne veut pas jouer sur une certaine scène où s'il ne veut pas jouer avec certains objets, cela sera paramétrable dans le menu d'ergonomie qui reste vide, pour le moment.

## V.B- Intégration de scènes

### V.B.1- Création de scènes

Sur le prototype existant existaient deux scènes : la chambre et la cuisine, seulement, ajouter de la diversité dans les environnements de niveaux ne peut qu'améliorer la qualité du jeu et l'expérience de l'enfant. C'est pourquoi je me suis attelé à la création de la scène du salon. Pour ce faire, il m'a été fourni un fichier graphique en 2D isométrique (toutes les scènes du jeu sont en 2D isométrique) avec une grande partie voire tous les éléments graphiques dont j'avais besoin. Il m'a aussi été donné de la documentation sur Affinity Designer et Unity, et créer une scène de jeu pour *Escape It*. Pour créer la scène salon, il faut exporter les différents éléments de la scène graphique et utiliser ces exports en tant que sprites (image, apparence d'un GameObject dans Unity) des prefabs (GameObjects personnalisés instanciables dans Unity, ce qui est utile pour nous car les scènes de jeu sont générées dans Unity depuis un fichier xml à l'aide de code, d'où la nécessité de faire des prefabs pour tous les éléments de la scène). Ensuite, en fonction du rôle de chaque prefab, il fallait leur attribuer le bon monobehaviour (script en c# qui décrit le fonctionnement d'un GameObject dans Unity) avec le bon paramétrage, ce qui est particulièrement fastidieux. Certains objets sont des éléments de solution, qui sont bougeable à l'aide d'un script de drag and drop (que je n'ai pas écrits, comme tous les *monobehaviours* des prefabs de scène de jeu d'*Escape It*), d'autres sont aussi des objets que l'on peut changer de place mais qui ne permettent pas de résoudre l'énigme, tandis que d'autres sont des éléments de décor etc (voir figure 2).

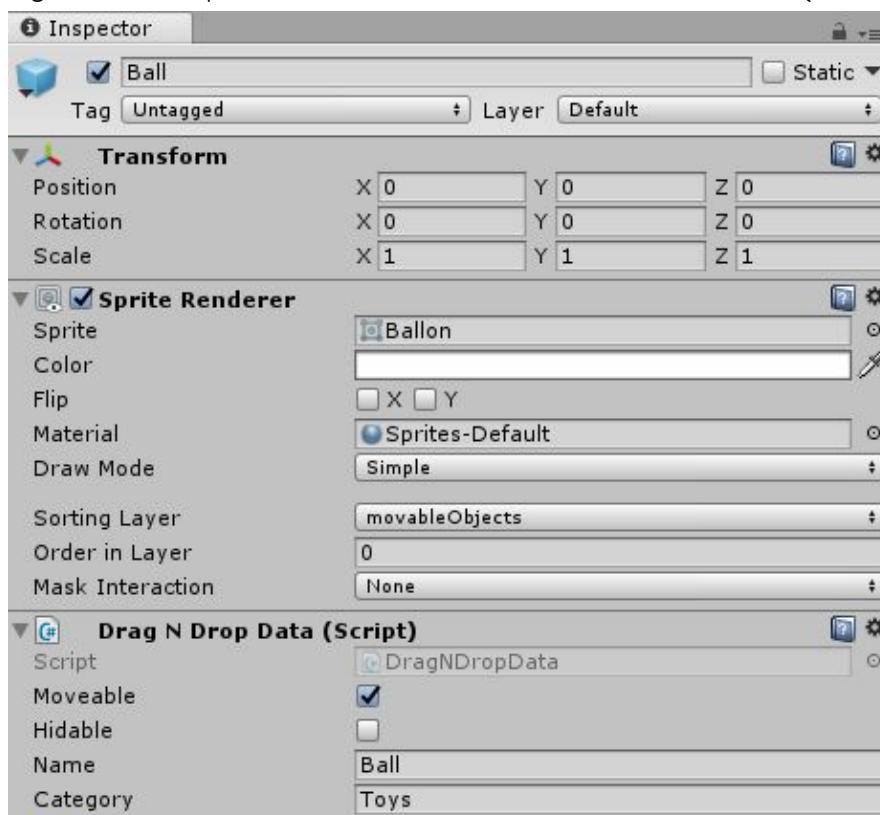


Figure 17 : Paramétrage du prefab Ball

## **V.B.2- Intégration de scènes existantes dans un autre projet Unity**

Quand M.Laforcade modifiait des scènes et voulait que je les intègre dans mon projet Unity alors je prenais une copie de son projet. Ainsi, en exportant les scènes et les prefabs et en les important dans mon projet Unity m'aurait rendu la tâche facile si l'export/import gardait en mémoire les références : ce qui n'est pas le cas. Il a donc été nécessaire de réaffecter tous les scripts manquants à tous les prefabs et les paramétrer correctement, ainsi que régler toutes les références aux layers (Ce qui permet à Unity de savoir quels éléments afficher devant et lesquels afficher derrière dans une scène de jeu en 2D).

## **V.C- Tâches diverses**

### **V.C.1- Communication Unity/Générateur**

Comme mentionné précédemment, avant mon arrivée sur le projet existait le générateur de scénarios de jeu, cependant, il n'était pas intégré au prototype existant, ce pourquoi une de mes tâches était d'intégrer l'appel au service de génération de scénarios adaptés en ligne. Pour faire fonctionner cette communication entre Unity et le générateur, il nous fallait une technologie assez simple et déployable rapidement, et nous avons décidé d'utiliser un script php. Pour comprendre l'appel au générateur, il faut comprendre le processus de génération. Pour générer un scénario, le générateur a besoin d'informations capitales :

- Le nombre de scènes de jeu à générer
- Le statut de l'activité (active ou non) de chaque compétence du profil
- Le statut de l'acquisition (acquise ou non) de chaque compétence du profil
- Le niveau de maîtrise de chaque compétence du profil

Si le générateur a besoin de telles informations sur les différentes compétences du profil, c'est pour savoir s'il ne doit pas générer des scènes en rapport avec certaines compétences (d'où la nécessité du statut de l'activité ou de l'acquisition des compétences), à quelle difficulté doivent appartenir les niveaux qu'il génère dans chaque compétence, sans oublier qu'un niveau de maîtrise élevé dans une compétence ou son acquisition permet de débloquent de nouvelles compétences à travailler. Tout ceci permet d'améliorer l'expérience de jeu de l'enfant avec une difficulté progressive qui s'adapte à sa vitesse d'apprentissage.

Le générateur aura besoin, pour fonctionner, que l'on lui donne un fichier xml avec ces informations suivant une structure déjà prédéfinie. Notre premier rôle est donc, dans Unity, de générer ce fichier en respectant son contenu et sa structure. Nous remarquerons qu'à part le nombre de niveau à générer dans le scénario, les informations nécessaires à la génération sont contenues dans le profil, et, pour obtenir cette information, il suffit de réaliser un écran dans lequel on demande à l'utilisateur (voir écran de choix du nombre de niveaux sur la figure 4). Une fois toutes les informations réunies, on crée une classe sérialisable avec les attributs correspondant avec ces informations, et on utilise les fonctionnalités de XMLSerializer de Unity pour paramétrer

le nom des balises afin que la structure globale soit celle dont a besoin le générateur. Enfin on sérialise les informations et on crée (ou écrase s'il existe déjà) le fichier d'entrée du générateur.

Le processus de génération peut être décomposé en trois étapes. La première étape est la traduction du fichier xml d'en un format de données interprétables par le générateur, la seconde est la génération du scénario en elle-même, et la dernière est la création d'un fichier xml correspondant au scénario généré et interprétable par le *LoaderGameManager* d' *Escape It*

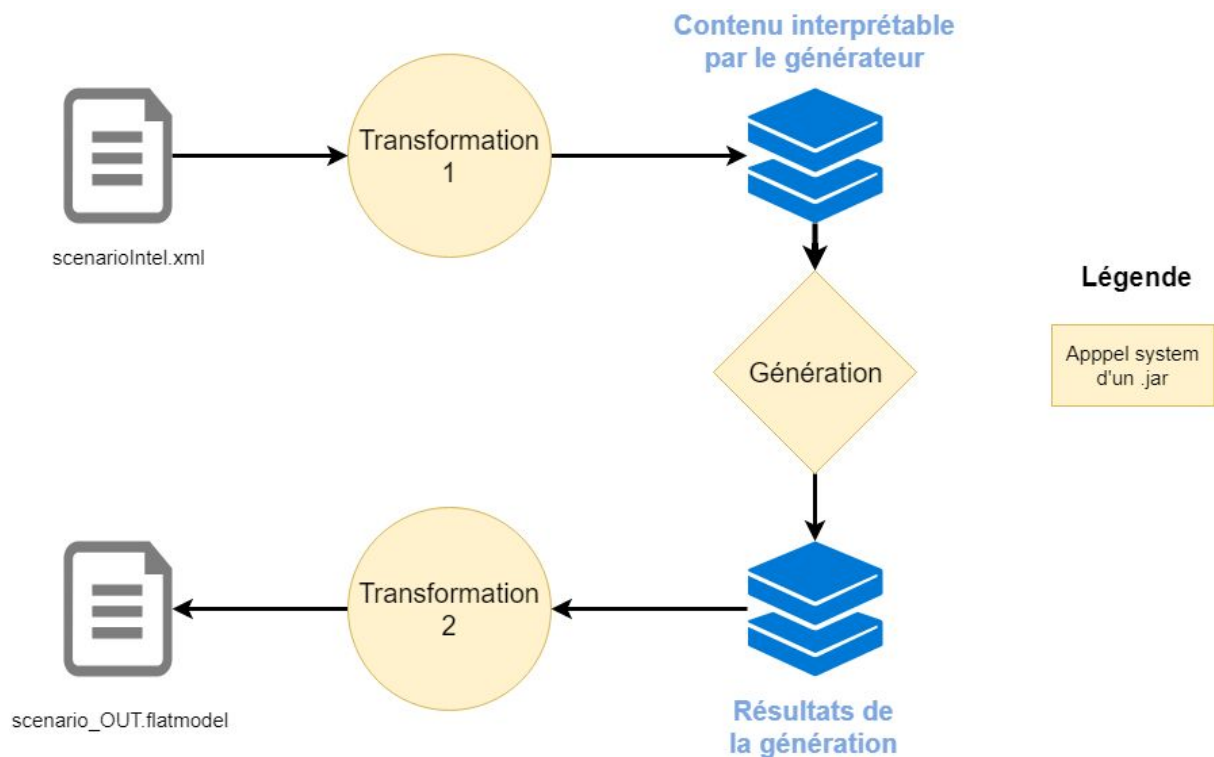


Figure 18 : Schéma du processus de génération

Pour effectuer la connexion entre le jeu et le service de génération de scénarios adaptés, il a donc fallu envoyer les informations sérialisées vu ci-dessus et récupérer le contenu du fichier créé par le générateur pour instancier les scènes de jeu dans Unity.

L'envoi des informations au générateur se fait, dans un premier temps, par le biais de la lecture du fichier et de l'envoi depuis Unity d'une requête POST sur une adresse dédiée au générateur avec le contenu du fichier, à l'aide de la classe WWW de Unity. C'est là que s'exécute mon script php (voir figure 19) dont le fonctionnement est simple :

- On vérifie si un contenu a été posté
- On récupère le contenu posté
- On crée ou écrase le fichier `scenarioIntel.xml` avec le contenu posté, à un endroit précis dans l'arborescence où le générateur viendra le chercher
- On fait un appel à un shell script qui fait des appels système qui exécutent les différentes parties du processus de génération

- On récupère le contenu de *scenario\_OUT.flatmodel* qui est le résultat de la génération et on l'affiche à l'aide d'un *echo*.

Lorsque l'on envoie une requête via un objet *WWW* de Unity, celui-ci possède une fonctionnalité qui permet à l'utilisateur d'attendre une réponse et donc ici, l'arrêt de l'exécution du script php. ainsi, une fois que l'objet *WWW* a reconnu le fait qu'une réponse a été envoyée, on récupère le contenu de cette réponse (qui n'est autre que le contenu du *scenario\_OUT.flatmodel*, le résultat de la génération) et on l'enregistre dans les fichiers locaux, en créant ou en écrasant (une nouvelle fois) le fichier.

Pendant tout ce processus, l'utilisateur, lui est redirigé vers un écran de chargement, qui, lorsque le scénario est lu et généré dans Unity par le *LoaderGameManager*, est à nouveau redirigé, cette fois-ci vers l'écran de la structure du jeu (voir figure 4).

```
<?php
    if(isset($_POST['xml'])){
        header("Content-type: text/xml");
        //[OK]
        //Mise en position du fichier scenarioIntel.xml
        $path = getcwd();//récupère la position absolue du dossier parent
        du fichier Index.php à savoir le dossier EscapeIt
        $myfile =
fopen("/var/www/html/escapeit/Generator/profile/scenarioIntel.xml", "w") or
die("Unable to open scenarioIntel.xml!");
        $txt = $_POST['xml'];
        fwrite($myfile, $txt);
        fclose($myfile);
        $output=shell_exec('/bin/sh
/var/www/html/escapeit/Generator/lance.sh');
//shell_exec('sudo -S sh /var/www/html/escapeit/Generator/lance.sh');
//exec('start '.$path.'\Generator\lance.bat');
//exec('cd '.$path.'\Generator & start lance.bat');

        $myfile = fopen("/var/www/html/escapeit/Generator/error.txt", "r")
or die("Unable to open error.txt");
        if(filesize("/var/www/html/escapeit/Generator/error.txt")>0){
            $error =
fread($myfile,filesize("/var/www/html/escapeit/Generator/error.txt"));
        }
        fclose($myfile);

        if(empty($error)){
            //[OK]
            //Renvoi à Unity du Scenario_OUT.flatmodel

            $myfile =
fopen("/var/www/html/escapeit/Generator/Scenario_OUT.flatmodel", "r") or
die("Unable to open Scenario_OUT.flatmodel!");
```

```
        echo
fread($myfile,filesize("/var/www/html/escapeit/Generator/Scenario_OUT.flatmodel"
));
        fclose($myfile);

    }else{
        echo $error;
    }

}else{
    echo "xml not posted";
}

?>
```

Figure 19 : script php d'envoi des informations et exécution de la génération

### V.C.2- Sécurité enfant

Si l'utilisation d'*Escape It* par l'enfant est supposée être supervisée par son thérapeute, il est quand même nécessaire de restreindre son accès aux parties du jeu qui ne lui sont pas destinées, qui sont, par ailleurs, au nombre de deux : les options et l'écran de la guidance (voir figure 4). L'écran de la guidance intervient à la fin de chaque niveau du scénario et se décompose en trois phases :

- L'enfant donne la tablette au thérapeute
- Le thérapeute indique si l'enfant a effectué le niveau seul ou non
- Le thérapeute rend la tablette à l'enfant

Lorsque l'enfant arrive sur l'écran de la guidance, une interface s'affiche et lui indique qu'il doit donner la tablette au thérapeute. Si dans les préférences du profil, le code est enregistré et différent de "0000", alors cet écran affichera un digicode (voir figure 20), avec lequel il faudra saisir le code des préférences pour passer à la deuxième phase. Cette étape est nécessaire, car il est important que l'enfant ne voit pas ce que fait le thérapeute, ce qui pourrait être source de frustration dans le cas inverse.

L'enfant n'a aucun intérêt à voir les options (à part peut-être pour changer son avatar). C'est pour cela que l'accès aux options est protégée de la même manière. Ce code peut être modifié dans les options dans l'onglet préférences (voir figure 13), en appuyant sur le bouton de modification. Une interface semblable à la figure 20 s'affiche et il suffit d'y saisir le code désiré. Lors de la création du profil, ce code a la valeur par défaut "0000", avec laquelle les digicodes ne s'affichent pas.

Le code est un code à 4 chiffres. Le digicode (figure 20) permet de visualiser combien de chiffres ont été entrés, d'effacer le dernier chiffre (la flèche vers la gauche),

et même de tout effacer (bouton “clear”). Lorsque le code est correctement entré, le digicode disparaît et l’écran d’après s’affiche (pas besoin de valider le code entré avec un bouton). La croix rouge en haut à droite permet de faire disparaître le digicode sans avoir à le remplir mais ne redirige pas vers les écrans protégés. Cette croix n’existe pas sur l’écran de la guidance.

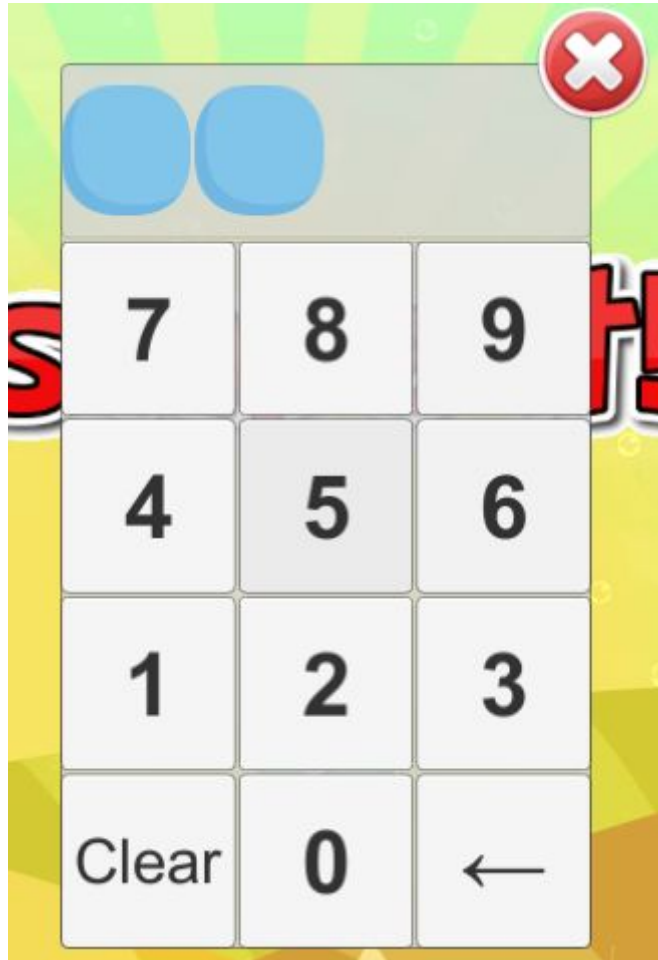


Figure 20 : Interface du verrou

### V.C.3- gestion des fichiers locaux

*Escape It* a besoin à plusieurs reprises de gérer des fichiers, que ce soit pour sauvegarder les profils et les préférences ou de lire des scénarios. Toute cette gestion de fichiers se fait en local et a besoin de respecter une structure (voir figure 21) par nécessité d’ordre et de clarté. La gestion explicite des fichiers locaux se fait dans le dossier *XMLFiles*, et on trouve à sa racine la sauvegarde des profils, le fichier d’entrée du générateur et le scénario généré, ainsi que deux dossiers, dans lesquels sont sauvegardées les informations liées aux préférences et aux renforçateurs. La structure du nom de ces fichiers de sauvegarde est importante et fonctionne ainsi : pour les préférences, on a : “child”+identifiant du profil+”preferences.xml” tandis que pour les renforçateurs on a : reinforce+identifiant du profil+thème du renforçateur + “Save.save”.



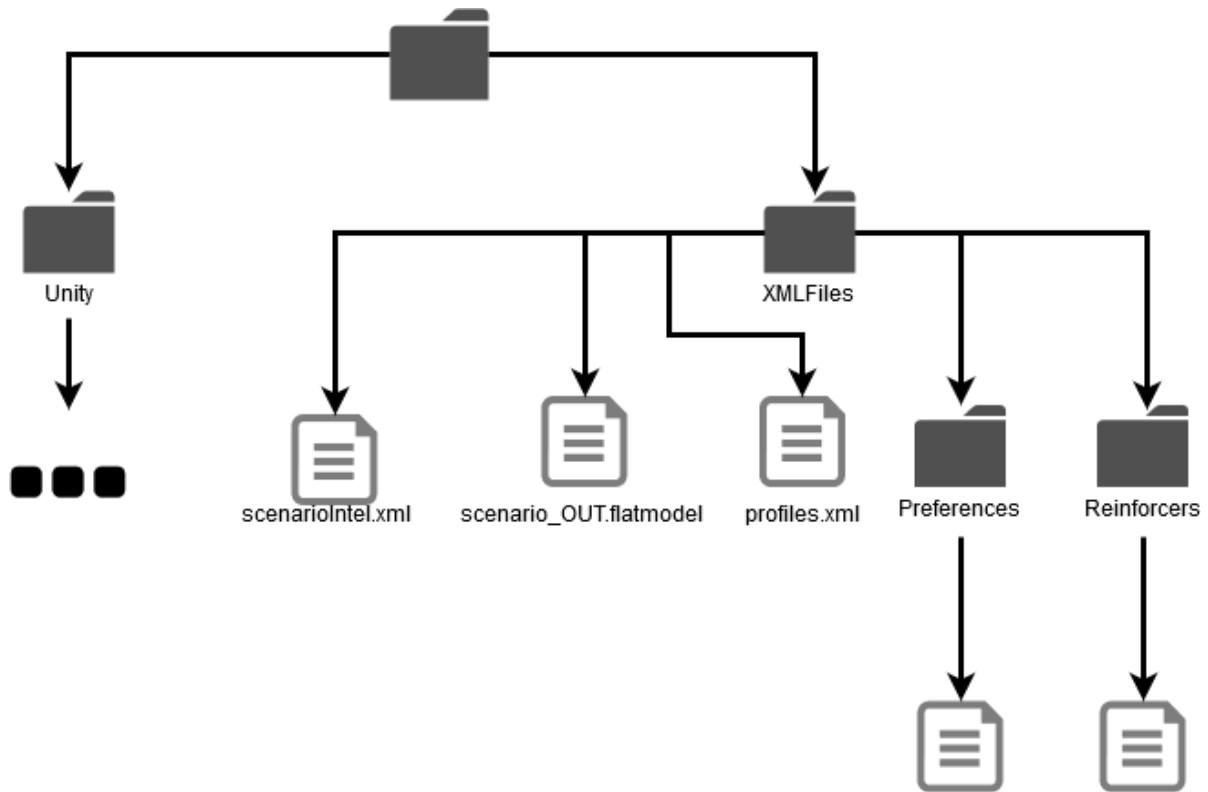


Figure 21 : Arborescence des fichiers locaux du projet

#### V.C.4- création d'images

Durant la réalisation de toutes mes tâches, la plupart ont nécessité la créations d'écrans et interfaces utilisateur qui demandent en général des boutons, des images etc. Cependant, toutes les images et tous les boutons dont j'avais besoin n'ont pas toujours été présents dans le projet Unity et, jusque là, nous n'utilisions que les images fournies dans un paquetage de ressources gratuit pour Unity avec un style graphique particulier, ce pourquoi j'ai dû avoir recours à Gimp. En effet j'ai utilisé les boutons existants (le bouton play par exemple) afin d'en refaire de nouveaux (le bouton de suppression par exemple) avec des icônes totalement libres de droit (sans aucune mention) depuis le site <https://www.freepik.com/>.

## VI- Ce que le stage m'a apporté

Pendant ce stage, j'ai appris de nombreux aspects de la recherche, dans le cadre de l'informatique en tous cas. En effet, en travaillant à proximité de doctorants, post-doctorants et maîtres de conférences, j'ai pu affiner ma vision du monde de la recherche et du parcours pour y parvenir et y progresser. Même si tous ces aspects du monde de la recherche ne m'ont pas forcément attiré (et certains m'ont même repoussé), ce stage m'a conforté dans mon choix d'études et de parcours professionnel dans le monde de la recherche. Effectivement, en discutant avec d'autres membres de ma promotion, eux ayant vécu une expérience dans le monde de l'entreprise, je me suis rendu compte que malgré ses défauts, le monde de la recherche m'attirait plus (travailler au pôle recherche et développement d'une entreprise serait tout aussi satisfaisant pour

moi). Au final, ce stage m'a permis de grandir et d'acquérir plus de maturité en ce qui concerne mon projet professionnel (si ce n'est pas un domaine de métier parfait, c'est un domaine dont les qualités rendent les défauts négligeables de mon point de vue).

De plus, durant ce stage, j'ai pu découvrir de nombreuses et diverses applications de l'informatique qui est un domaine d'une polyvalence extrême, ce qui est d'ailleurs aussi le cas en entreprise, je n'en doute pas. Je pense d'ailleurs que l'informatique est un domaine indispensable pour tous les autres et c'est certainement ce qui fait que c'est un domaine en pleine expansion même s'il n'est plus tout récent, et qui a fait que ce stage me fasse penser travailler dans l'informatique est un pari sûr pour l'avenir.

Ce stage a aussi été l'occasion d'apprendre ou d'approfondir de nouvelles compétences dans le domaine de l'informatique comme la création et l'édition d'images (simples et vectorielles), la programmation en C#, la création de jeu avec le moteur Unity, quelques bases du Game Design, et même en montage vidéo. En dehors des compétences informatiques, ce stage m'a principalement appris à intégrer un projet en cours de route, ce qui peut être complexe tant il faut s'approprier le sujet, ce qui a été fait et ce qu'il reste à faire, qui sont les membres du groupes et quelles sont leurs missions, et en quoi ces missions rejoignent les miennes, etc.

La réussite de ce stage est globalement due aux compétences que j'ai apprises en deux années de DUT Informatique que j'ai su utiliser dans le contexte du projet. En effet j'ai eu recours à ces compétences en rapport avec le domaine de l'informatique mais pas seulement. J'ai par exemple utilisé mes compétences en programmation avancée (optimisation et complexité des algorithmes), en programmation orientée objet (comme l'impose le système de prefab de Unity et les règles de base du Game Design en général), la programmation en multithread (enchaînement des écrans et actions en arrière plan), en programmation web (communication entre Unity et le service de génération adapté en ligne), en Interface Homme-Machine (pour la réalisation des toutes les interfaces utilisateur et écrans) et en conception. Cependant, j'ai également utilisé des compétences en gestion de projet (division des tâches et organisation dans le temps) et en anglais (pour lire la documentation de Unity, regarder leurs tutoriels ou encore chercher des solutions à des problèmes sur des forums).

Pour finir, ce stage a également été l'occasion pour moi de découvrir les méthodes de travail en laboratoire et de faire évoluer les miennes. J'ai d'abord pu remarquer que dans les petites équipes, l'organisation est très malléable et prend la forme voulue par le responsable (dans l'exemple de mon stage sur le projet *Escape It*, nous faisons souvent de petites réunions, nous utilisons un trello, etc.) cependant il existe quand même dans le laboratoire des grandes réunions d'équipes (une fois par mois), mais aucune méthode de travail imposée à proprement parler. Pour autant, ce stage a fait évoluer mes méthodes de travail, dans le sens où j'ai découvert le fait de travailler en collaboration avec une personne tout en étant sous son autorité, j'ai appris à me coordonner avec des personnes dont le travail n'a que peu de lien avec le mien au sein du projet, et j'ai en plus appris à travailler en autonomie sur un projet long et complexe, ce qui demande une organisation plus rigoureuse que ce que j'ai pu faire avant.

## VII- Conclusion

Le but du projet *Escape It* est de faciliter l'apprentissage des enfants autistes. Les obstacles aux apprentissages des enfants autistes peuvent venir de leur difficultés à généraliser, imiter et déduire l'implicite. Certaines compétences apprises implicitement par des enfants neurotypiques doivent alors être travaillées pour les enfants autistes. L'acquisition, le renfort, et la généralisation de ces compétences sont généralement atteints avec un apprentissage à base d'essais-erreurs. Aujourd'hui, ces enfants ont des prédispositions pour les supports mobiles tactiles, ainsi, afin de faciliter le support de ces apprentissages, l'objectif est de développer un jeu sérieux de type Escape-the-room sur tablette.

Les aspects de recherche de ce projet ont été effectués auparavant et ne constituent pas mon sujet de stage. L'objectif final de ce stage est de pouvoir donner un prototype du jeu sérieux à des thérapeutes pour avoir leurs retours. Il y a un déjà un prototype existant, cependant, il manque de fonctionnalités importantes qui font l'objet de mon stage.

Durant toute ma période de stage, j'ai pu concevoir, développer et tester les fonctionnalités suivantes :

- implémenter la gestion de profils
- gérer la sauvegarde des profils et préférences
- intégrer l'appel au service de génération de scénarios adaptés en ligne
- créer un niveau du jeu
- gérer la progression du joueur dans les différentes compétences
- réaliser les interfaces utilisateur nécessaires
- faire du refactoring et des améliorations sur le code déjà existant

L'implémentation des profils permet au jeu d'adapter les scénarios aux progrès des enfants, et l'implémentation de la sauvegarde permet de ne pas perdre cette progression d'une session à l'autre. La communication avec le service de génération de scénarios adaptés permet la non redondance des scènes qui pourraient s'avérer inefficaces sur le long terme, tout comme la création de niveaux supplémentaires. Toutes ces modifications étaient indispensables à *Escape It* afin de pouvoir être testé et même utilisé par les thérapeutes.

Pour améliorer encore le projet, il faudrait rajouter la gestion d'autres compétences visuelles sur la scène du salon, implémenter la sauvegarde de traces d'actions des sessions de jeu, et résoudre des potentiels problèmes de résolution sur un support iOS ou problèmes mineurs dans les scènes.

Le projet a pour but d'être un prototype testé par des thérapeutes, ce sont leur remarques et retours qui feront évoluer le projet en espérant que celui-ci dépasse le stade de prototype et puisse un jour aider de nombreux enfants qui en ont besoin.

## VIII- Glossaire

LIUM : Laboratoire d'Informatique de L'université du Maine.

LST : (Language and Speech Technology ou Reconnaissance de la parole et traduction automatique)

IEIAH : Ingénierie des Environnements Informatiques pour l'Apprentissage Humain

ABLLS-R : Wikipédia nous apporte cette définition : «L'évaluation des aptitudes langagières et d'apprentissage de base est un outil pédagogique fréquemment utilisé pour l'analyse comportementale appliquée (Applied Behaviour Analysis) pour mesurer les compétences linguistiques et fonctionnelles de base d'une personne ayant des retards de développement ou des handicaps.»

([https://en.wikipedia.org/wiki/Assessment\\_of\\_basic\\_language\\_and\\_learning\\_skills](https://en.wikipedia.org/wiki/Assessment_of_basic_language_and_learning_skills))

## IX- Références bibliographie/webographie

Site du LIUM : <https://lium.univ-lemans.fr/lium/>

Site d'Escape It : [https://projets-lium.univ-lemans.fr/Escape It/](https://projets-lium.univ-lemans.fr/Escape%20It/)

FreePik : <https://www.freepik.com/>

Présentation "CSEDU'18 - ESCAPE IT!"

Présentation "Projet Escape It! - Evolution prototypes"