



HAL
open science

LUCIOLE: analyse et évaluation des effets d'un serious game et son apport pour l'apprentissage de l'anglais

Hesna Sebiane

► **To cite this version:**

Hesna Sebiane. LUCIOLE: analyse et évaluation des effets d'un serious game et son apport pour l'apprentissage de l'anglais. Sciences de l'Homme et Société. 2021. dumas-03537333

HAL Id: dumas-03537333

<https://dumas.ccsd.cnrs.fr/dumas-03537333v1>

Submitted on 20 Jan 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



LUCIOLE : analyse et évaluation des effets d'un *serious game* et son apport pour l'apprentissage de l'anglais

Hesna SEBIANE

Sous la direction de Claude PONTON

Laboratoire : LIRIS / ICAR

UFR LLASIC
Département I3L

Mémoire de master 2 mention Sciences du langage - 20 crédits

Parcours : Industrie de la Langue

Année universitaire 2020-2021

LUCIOLE : analyse et évaluation des effets d'un *serious game* et son apport pour l'apprentissage de l'anglais

Hesna SEBIANE

Sous la direction de Claude PONTON

Laboratoire : LIRIS / ICAR

UFR LLASIC
Département I3L

Mémoire de master 2 mention Sciences du langage - 20 crédits

Parcours : Industrie de la Langue, orientation professionnelle

Année universitaire 2020-2021

Remerciements

Je saisis cette occasion pour adresser mes remerciements à l'ensemble des enseignants du Master Industrie de la Langue de l'Université Grenoble Alpes pour la qualité de l'enseignement et particulièrement à Claude Ponton, qui est aussi mon tuteur de formation, pour son suivi et ses conseils précieux pendant ces deux dernières années. L'écoute et l'accompagnement dont j'ai bénéficié m'ont permis d'affiner mon projet professionnel et de réaliser ce stage. Je tiens aussi à remercier Thomas Lebarbé d'avoir accepté de faire partie de mon jury.

Je souhaiterais remercier tout particulièrement mon tuteur de stage, Mathieu Loiseau, pour sa confiance et les connaissances qu'il a su partager avec moi. Je le remercie aussi pour sa disponibilité et la qualité de son encadrement.

Je remercie également ma co-tutrice de stage, Émilie Magnat, pour son accueil à ICAR et sa gentillesse. Merci aux membres de l'équipe pédagogique du projet, Coralie Payre-Ficoult et Marie-Pierre Jouannaud, pour les échanges riches que nous avons eus sur le jeu LUCIOLE.

Pour terminer, je tiens à remercier mes parents pour leur soutien constant et leurs encouragements.

DÉCLARATION ANTI-PLAGIAT

1. Ce travail est le fruit d'un travail personnel et constitue un document original.
2. Je sais que prétendre être l'auteur d'un travail écrit par une autre personne est une pratique sévèrement sanctionnée par la loi.
3. Personne d'autre que moi n'a le droit de faire valoir ce travail, en totalité ou en partie, comme le sien.
4. Les propos repris mot à mot à d'autres auteurs figurent entre guillemets (citations).
5. Les écrits sur lesquels je m'appuie dans ce mémoire sont systématiquement référencés selon un système de renvoi bibliographique clair et précis.

PRENOM : Hesna.....

NOM : SEBIANE.....

DATE : 25/10/2021.....

Sommaire

Remerciements	3
Sommaire.....	5
Introduction	7
Partie 1 - Présentation du stage et contexte d'accueil	9
CHAPITRE 1. CONTEXTE D'ACCUEIL	10
1. Le Laboratoire d'InfoRmatique en Image et Systèmes d'information (LIRIS) ..	10
2. Le laboratoire Interactions, Corpus, Apprentissages, Représentations (ICAR) ..	11
CHAPITRE 2. PRESENTATION DU PROJET	12
1. Le projet Fluence e-FRAN	12
2. Les applications du projet Fluence	12
3. Le protocole expérimental du projet Fluence	14
CHAPITRE 3. OBJECTIF DU STAGE	16
1. Le stage : le projet LUCIOLE	16
2. Missions et tâches	16
Partie 2 - Le jeu LUCIOLE Conception, Fonctionnalités et données collectées	18
CHAPITRE 4. LES FONCTIONNALITES DU JEU LUCIOLE.....	20
1. Se connecter à LUCIOLE.....	20
2. Structure du jeu LUCIOLE.....	20
CHAPITRE 5. LES DONNEES COLLECTEES	26
1. Les mediaWiki.....	26
2. Tests et expérimentations	27
3. Définition des concepts utilisés	28
Partie 3 - Analyse des traces LUCIOLE.....	33
CHAPITRE 7. LA PRISE EN MAIN DES TRACES.....	34
1. Modélisation des traces LUCIOLE	34
2. Exportation des traces LUCIOLE	36
3. Formalisme des traces en JSON	38
CHAPITRE 8. TRANSFORMATION DES TRACES LUCIOLE	40
1. Exportation des traces XML en JSON	40
2. Extraction des traces EVASION et ELARGIR	40
CHAPITRE 9. DESAMBIGUÏSATION DES TRACES LUCIOLE	41
1. Fusion des indicateurs	41
2. Réécriture des traces	42
3. Gestion des sessions de jeu.....	44

CHAPITRE 10. CALCUL DES INDICATEURS	46
1. Le constructeur CSV	46
2. Création des routines pour le calcul des indicateurs.....	46
3. Récapitulatif du travail réalisé et les difficultés rencontrées	49
4. Perspectives de l'analyse des traces LUCIOLE et extensions possibles.....	50
Conclusion et bilan personnel.....	53
Bibliographie	55
Sitographie.....	56
Sigles et abréviations utilisés.....	57
Table des illustrations	58
Table des annexes	59
Table des matières	66

Introduction

L'avènement du numérique a induit des changements significatifs dans la pédagogie et a conduit à des évolutions du métier de l'enseignant en introduisant une dimension ludique de l'apprentissage, en particulier l'apprentissage d'une langue étrangère. Nous assistons aujourd'hui, de plus en plus à des conceptions innovantes de l'apprentissage des langues étrangères, engendrées par le développement de l'informatique et fortement influencées par l'essor des jeux vidéo. Ce lien entre le jeu numérique et l'apprentissage ou l'éducation a fait émerger la notion de *serious game* ou jeu sérieux.

Le jeu sérieux est cette hybridation entre le divertissement et l'apprentissage ou « ludo-éducative » (Bourgère, 2012), qui met en place des mécanismes ludiques pour rendre un apprentissage plus attrayant à des fins précises. Si la conception et les usages des jeux sérieux restent discutables en raison du manque des protocoles expérimentaux, néanmoins ils peuvent constituer un support complémentaire à l'activité de l'enseignant.

Après de nombreuses interrogations sur la notion de jeu sérieux et son apport dans l'apprentissage d'une langue étrangère, une équipe de chercheurs de l'Université Grenoble Alpes a conçu un jeu sérieux, en s'appuyant sur les théories de l'acquisition basées sur l'usage (Bybee, 2008). Ayant constaté des difficultés de la compréhension orale de l'anglais chez les élèves français (Commission Européenne, 2012), l'équipe menant ce projet, s'est focalisée sur cette activité langagière pour concevoir le jeu LUCIOLE comme un dispositif autonome de l'apprentissage suivant sa progression dans le jeu.

Ce travail de fin d'études a donc été réalisé dans le cadre du projet LUCIOLE « LUdique au service de la Compréhension Orale en Langue Étrangère » avec Mathieu Loiseau, (porteur du projet), membre de l'équipe SICAL du laboratoire LIRIS et Émilie Magnat (co-responsable du projet), membre du laboratoire ICAR. Le jeu LUCIOLE fait partie du projet e-FRAN Fluence, qui est porté par le CNRS et l'Université Grenoble Alpes. Destiné à des élèves de cycle 2, LUCIOLE est un jeu sérieux visant à développer la compréhension orale en anglais, en introduisant une dimension ludique et narrative dans le jeu.

A l'issue d'une étude longitudinale de cohortes d'élèves du CP au CE2 dans les académies de Grenoble et de Mayotte, LUCIOLE a montré des résultats encourageants suite

aux premières analyses des pré et post-tests réalisées au début de chaque année d'expérimentation. Désormais, nous tenterons, dans ce qui suit, d'analyser les traces LUCIOLE afin de mettre en lien les effets du système avec les pré et post-tests.

Pour mener à bien ce travail, nous présenterons, en premier lieu le stage réalisé et le contexte d'accueil. Nous tenterons ensuite, d'expliquer la conception du jeu LUCIOLE et ses différentes fonctionnalités, ainsi que les données collectées dans le cadre de ce projet. Enfin, nous clarifierons le travail effectué pour l'analyse des traces LUCIOLE et nous proposerons quelques perspectives pour mesurer les effets du jeu avec les pré et post-tests.

Partie 1

-

Présentation du stage et contexte d'accueil

Chapitre 1. Contexte d'accueil

1. Le Laboratoire d'InfoRmatique en Image et Systèmes d'information (LIRIS)

Le Laboratoire d'InfoRmatique en Image et Systèmes d'information (LIRIS) regroupe 330 membres provenant de différents établissements dont le CNRS, l'INSA (Institut National des Sciences Appliquées) de Lyon, l'Université Claude Bernard Lyon 1, l'Université Lumière Lyon 2 et l'Ecole Centrale de Lyon. Les recherches menées au LIRIS ont pour objet principal la recherche et le développement en sciences informatiques.

Grâce à la diversité de ses activités de recherche dans différents domaines comme l'ingénierie, la médecine, les sciences de la vie, les sciences humaines et sociales et les sciences de l'environnement, le LIRIS cherche à mettre en exergue les dispositifs numériques et les jalons de l'Intelligence Artificielle (IA). Le laboratoire LIRIS se veut impliqué dans les défis sociétaux et le développement durable. Il se compose de six pôles de compétences et réunit douze équipes de recherche comme illustré dans la figure ci-dessous :

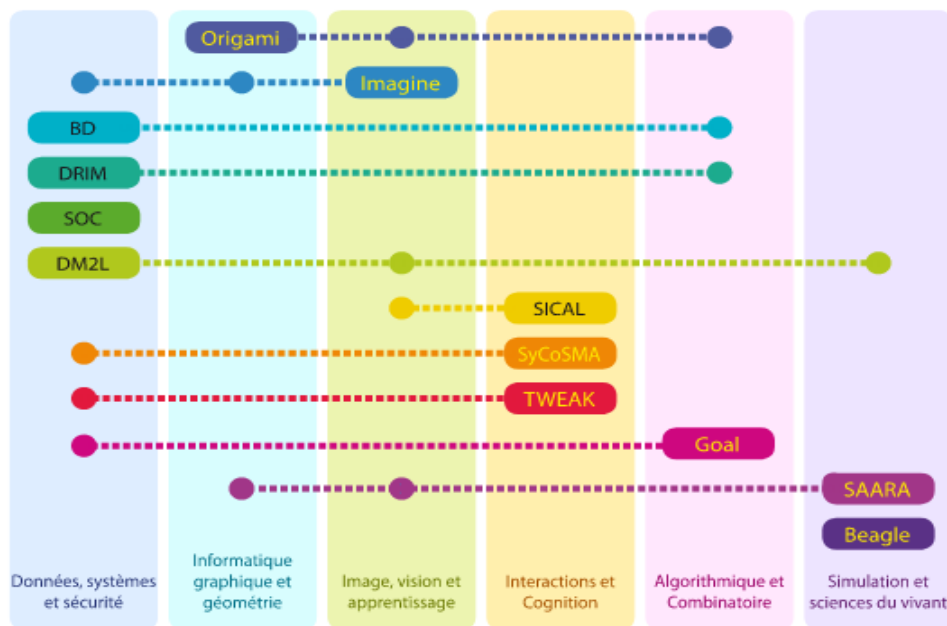


Figure 1. Les équipes de recherche par pôle de compétence¹

¹ <https://liris.cnrs.fr/liris>

2. *Le laboratoire Interactions, Corpus, Apprentissages, Représentations (ICAR)*

Le laboratoire Interactions, Corpus, Apprentissages, Représentations a été créé en 2003 et relève de l'Institut des Sciences Humaines et sociales du CNRS, de l'Université Lumière Lyon 2 et de l'ENS de Lyon. Il est spécialisé dans le domaine de l'interaction et s'intéresse essentiellement aux langues européennes et à l'arabe. En effet, ses recherches pluridisciplinaires en linguistique et en didactique étudient la langue en interaction et dans le texte dans sa multidimensionnalité.

Les recherches à ICAR ont permis de constituer des banques de données construites à partir de corpus multimodaux de grande taille. Afin de faciliter leur l'exploitation, ICAR a développé un moteur de requête permettant ainsi de proposer une plateforme, CLAPI (Corpus de Langue Parlée en Interaction) pour accéder à ses banques de données, ainsi qu'une plateforme pour l'analyse textométrique, appelée TXM. Il existe trois grands axes à ICAR, au sein desquels plusieurs thématiques y sont traitées par ses équipes : Équipe InSitu (INteractions et SITUations, pratiques et outils) ; Équipe ADIS (Apprentissages, Discours, Interactions, Savoirs) ; Équipe LanDES (Langue Discours Énonciation Sémiotique).

Chapitre 2. Présentation du projet

1. *Le projet Fluence e-FRAN*

Le projet Fluence² est l'un des 22 projets lauréats de l'action e-FRAN³, élaborée dans le cadre de la mission interministérielle sur le numérique éducatif et financée par le PIA (Programme d'Investissements d'Avenir) pour expérimenter l'utilisation du numérique dans les pratiques d'enseignement et d'apprentissage. Fluence est un projet soutenu par le CNRS et regroupe trois laboratoires de recherches de l'Université Grenoble-Alpes : LPNC (Laboratoire de Psychologie et NeuroCognition), GIPSA-Lab (Grenoble Images Parole Signal Automatique) et LIDILEM (Laboratoire de Linguistique et Didactique des Langues Etrangères et Maternelles) et celles des Académies de Grenoble et de Mayotte (Mandin et *al.*, 2021).

Il a pour objectif de concevoir des dispositifs numériques innovants pour optimiser l'apprentissage en classe. Fluence vise principalement l'apprentissage de la lecture et de l'anglais en tant que langue étrangère. Il cherche à

« [...] développer une lecture fluide et aisée via des entraînements spécifiques ciblant soit les mécanismes cognitifs sous-jacents (traitements visuels/visuo-attentionnels) soit des traitements holistiques inhérents à la lecture de textes.⁴ ».

Ce projet de recherche a permis la conception, le développement et la validation de trois applications sur tablette – EVASION, réalisée par le LPCN pour les traitements visuels et visuo-attentionnels impliqués en lecture – ELARGIR, développée au laboratoire GIPSA-Lab pour la lecture répétée et assistée de textes – LUCIOLE, pour la compréhension orale en anglais et est développée par le LIDILEM.

2. *Les applications du projet Fluence*

2.1. *EVASION*

EVASION vise à entraîner l'attention visuelle impliquée dans l'apprentissage de la lecture afin de faciliter la reconnaissance rapide des mots dans le but d'améliorer la lecture

² <https://fluence.prod.lamp.cnrs.fr/>

³ <https://www.gouvernement.fr/efran-les-22-laureats>

⁴ Voir la page du Wiki : <https://wiki.lezinter.net/ /LUCIOLE:Accueil#Acteurs>

chez l'enfant. Il s'agit de traiter des séquences de lettres de plus en plus longues et pendant des temps de plus en plus courts afin de mobiliser l'attention visuelle sur le mot à lire et d'être capable de traiter les lettres qui composent le mot. L'application se compose de quatre mini-jeux qui s'appuient sur le principe des jeux vidéo d'action. Au cours de la progression, EVASION varie selon les quatre critères suivants : la longueur des séquences, la similarité visuelle, le nombre de cibles simultanément présentées et la vitesse de défilement des séquences.

2.2. ELARGIR

ELARGIR est un outil basé sur le principe du karaoké et permet d'entraîner la fluence à partir de la lecture répétée ou assistée des textes. La lecture se fait à haute voix d'un texte en synchronie (syllabes, mots, groupes de sens, groupes de souffles, etc.) avec la voix d'un adulte pré-enregistrée. ELARGIR se fait en autonomie mais il sollicite également le travail en groupe pour la découverte et l'évaluation de la lecture. L'enfant procède à la lecture d'environ 10 à 15 textes courts, d'environ 200 à 300 mots ordonnés en difficulté croissante. Par ailleurs, le niveau de difficulté de la lecture (longueur du texte et style de lecture) augmente automatiquement dès lors que l'évaluation effectuée par un camarade est jugée satisfaisante. L'avantage d'ELARGIR est de permettre une meilleure mémorisation de l'orthographe, une accélération du traitement visuelle du texte et la prosodie en lecture.

2.3. LUCIOLE

LUCIOLE (LUdique au service de la Compréhension Orale en Langue Etrangère) est un jeu sérieux qui vise la compréhension orale en anglais. Il a été développé avec Unity (un moteur de jeu multiplateforme) et fonctionne sous iOS et Android. Il constitue un placebo pour les jeux EVASION et ELARGIR, car il ne fait pas appel aux modalités de jeu d'action⁵ et ne stimule pas les compétences de lecture chez l'enfant. Le jeu consiste à mettre en scène un jeune héros francophone qui se retrouve dans la peau d'un agent secret et devra se débrouiller en anglais afin de réaliser les différentes missions qui lui seront confiées. Dans son mémoire, Adrea Soh décrit le scénario du jeu LUCIOLE où :

⁵ Contrairement aux deux autres jeux du projet Fluence, EVASION et ELARGIR, le jeu LUCIOLE ne s'appuie pas dans sa conception sur le principe des jeux vidéo d'action, tels que la rapidité, le nombre d'actions à gérer, le défilement des objets (dans le temps et dans l'espace), etc. Voir la page du Wiki : https://wiki.lezinter.net/ /Jeu_d%27action

« Le joueur incarne Sasha, un Français d'environ 8-10 ans qui se fait recruter par un agent secret anglais (Ash). Au cours de plusieurs missions Sasha, accompagné de son mentor Ash et de l'entraîneur des espions (Hartguy), arrêtera des méchants qui ont enlevé des animaux de compagnie. Ils doivent partir en Angleterre et s'infiltrer parmi des méchants (anglophones) pour obtenir des informations importantes et libérer les animaux. C'est ainsi que Sasha devra acquérir et développer ses compétences en compréhension orale de l'anglais.⁶ ».

Par ailleurs, le jeu LUCIOLE s'appuie sur une histoire narrative qui débute en français, puis elle passe en anglais de manière progressive. Ainsi, le joueur est exposé aux différentes variétés de l'anglais au fil de ces voyages dans différents pays anglophones. Cela permet à ce dernier de découvrir de nombreux éléments culturels. Le jeu se focalise sur l'apprentissage du lexique, mais il propose également des activités de grammaire et de phonologie. Il favorise la répétition et l'entraînement systématique du lexique (grâce au carnet de notes). Une autoévaluation est intégrée dans le jeu afin de permettre à l'élève de prendre conscience de ses progrès et de développer des stratégies de communication dans une langue étrangère.

3. *Le protocole expérimental du projet Fluence*

Le projet Fluence se donne pour objectif de concevoir et expérimenter *in situ* les outils numériques pour l'apprentissage de la lecture et de l'anglais oral. Il propose alors des entraînements en classe réalisés sur tablettes numériques.

Pour mener à bien cette expérimentation, un protocole de suivi longitudinal de trois ans a été mis en place à partir de 2018 jusqu'à 2021, du fait de la pandémie du Covid-19 qui a empêché le bon déroulement de l'année 3 (c'est-à-dire l'année 2020). Pour cela, des cohortes d'élèves dans les académies de Grenoble et de Mayotte ont participé aux tests des trois logiciels du projet Fluence. Cette étude longitudinale a été testée sur 500 élèves du cycle 2, du CP au CE1 ou CE2 dans les deux académies pendant trois ans afin de prévenir les difficultés d'apprentissages d'une part, d'autre part, une cohorte de 200 élèves de 6^{ème} et de 5^{ème}, se trouvant en difficulté d'apprentissage, a participé au protocole Fluence durant deux ans, dans un objectif de remédiation. Ci-dessous, le plan d'expérimentation tel qu'il a été conçu par les équipes du projet Fluence :

⁶ <https://fluence.prod.lamp.cnrs.fr/luciole/> – Voir aussi le mémoire d'Adrea Soh qui se trouve dans les références de la page du Wiki du projet LUCIOLE : https://wiki.lezinter.net/w/images/775-m%C3%A9moire_M1_DILIPEM-Adrea_Soh.pdf

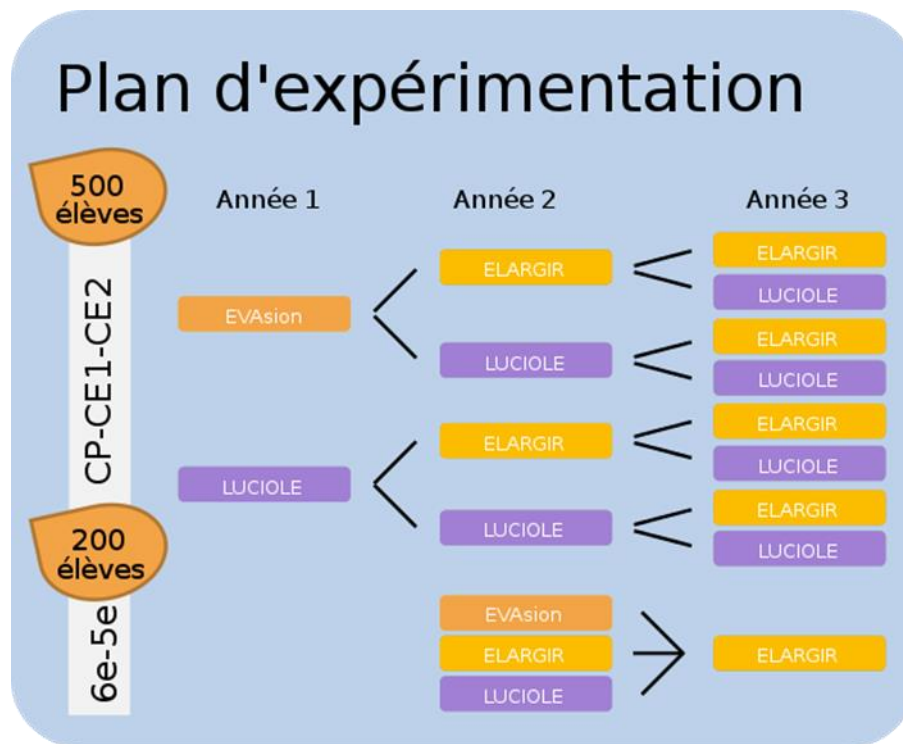


Figure 2. Plan d'expérimentation du projet Fluence⁷

Pour ce faire, des pré-tests ont été réalisés sur les élèves ayant participé à cette étude. Ces pré-tests ont permis aux différentes équipes travaillant sur les jeux Fluence de définir les compétences de lecture et de compréhension orale en anglais chez les sujets concernés et par conséquent, de former des groupes EVASION ou ELARGIR d'un côté, et LUCIOLE de l'autre. Il s'agissait de s'entraîner soit sur un jeu de lecture (EVASION⁸ ou ELARGIR) soit sur le jeu LUCIOLE pour travailler l'anglais oral. Des sessions d'entraînements dans les classes étaient consacrées aux différents jeux Fluence pendant 10 semaines (à raison de 3 séances de 20 minutes par semaine) aux élèves.

⁷ https://wiki.lezinter.net/_/Exp%C3%A9s:LUCIOLE/Plan_d%27exp%C3%A9rimentation

⁸ A noter que le logiciel EVASION n'est utilisé que pour les élèves du CP. Quant aux CE1 et CE2, ils se sont entraînés avec le logiciel ELARGIR. Cependant, LUCIOLE utilise des versions adaptées pour le niveau des élèves, allant du CP au CE2.

Chapitre 3. Objectif du stage

1. Le stage : le projet LUCIOLE

Bien que le projet Fluence e-FRAN soit porté par le CNRS et est sous la responsabilité de Sylviane Valdois (du laboratoire LPCN), celui-ci demande la collaboration de plusieurs équipes de recherche et le soutien de nombreuses institutions. C'est pourquoi, le Service des Langues de l'UGA (Université Grenoble Alpes) et le laboratoire LIDILEM s'est chargé de la création, de la gestion et du suivi du projet LUCIOLE. En effet, LUCIOLE est le produit de nombreux contributeurs (chercheurs, développeurs, stagiaires, acteurs, etc.). Toutefois, il est représenté par une équipe de chercheurs permanente chargée de la coordination du projet : Mathieu Loiseau, porteur du projet (LIRIS) avec Émilie Magnat, co-responsable du projet (ICAR), Coralie Payre-Ficout (LIDILEM) et Marie-Pierre Jouannaud (LIDILEM), co-responsables pédagogiques.

Pour autant, la validation du jeu LUCIOLE nécessite une analyse fine des traces, qu'il convient de croiser avec les résultats obtenus aux post-tests. De ce fait, un financement complémentaire a été accordé au projet LUCIOLE afin d'analyser les traces collectées. C'est ainsi qu'un stage a été financé par le Labex ASLAN de l'Université de Lyon et constitue en conséquence, l'objet de ce travail de fin d'études.

2. Missions et tâches

L'objectif même de notre travail consiste à analyser les traces LUCIOLE, recueillies à partir des entraînements des élèves ayant participé au protocole expérimental de Fluence pendant les trois années. Cette analyse a pour but de lier les traces LUCIOLE avec les résultats obtenus aux pré et post-tests. Durant cette période, les tâches que nous avons réalisées ont été réparties en quatre temps :

- La prise en main des données pré et post-tests et l'exportation des traces LUCIOLE à partir des fichiers XML.
- Premier niveau de transformation des traces : transmettre une liste de fichiers XML au programme afin de créer un fichier JSON par utilisateur en y agglomérant toutes ses traces, puis extraire et restructurer les informations.
- Restructuration et désambiguïsation des traces.

- Deuxième niveau de transformation des traces : définir les indicateurs pour calculer la progression de chaque utilisateur pour chaque année de test.

Partie 2

-

Le jeu LUCIOLE

Conception, Fonctionnalités et données collectées

Des études réalisées par la Commission Européenne en 2012⁹ ont montré que la compétence de la compréhension orale semble poser problème aux élèves français par rapport aux autres élèves européens. Cherchant à concevoir un jeu sérieux (Brougère, 2012) pour l'apprentissage d'une langue étrangère avec une dimension ludique, la conception du jeu LUCIOLE repose sur les théories basées sur l'usage (Bybee, 2008)¹⁰. L'intégration de LUCIOLE en classe n'est en aucun cas une activité de substitution à l'activité normale de l'enseignant, mais il propose plusieurs types d'activités, présentés sous forme de mini-jeux au sein d'un ensemble de missions. Les activités sont organisées de façon à traiter des contenus langagiers et permettre aux élèves de s'entraîner sur la compréhension orale en anglais.

Dans cette partie du travail, nous allons voir la structure et les fonctionnalités du jeu LUCIOLE, car cela nous aidera à mieux comprendre les traces collectées afin de les analyser. Nous évoquerons également les données des pré et post-tests recueillies dans le cadre de ce projet.

⁹ http://ec.europa.eu/dgs/education_culture/repository/languages/policy/strategic_framework/documents/languagesurvey-final-report_en.pdf

¹⁰ Les théories fondées sur l'usage telles qu'elles ont été conçues par Bybee (2008) « *reconnaissent l'impact de l'usage sur la représentation cognitive de la langue* (Bybee, 1985, 2001, 2006 ; Langacker, 1987, 2000) ». Cette nouvelle conception du langage accorde une importance particulière à la notion de l'usage qui repose sur l'input langagier, et dans laquelle la grammaire est considérée comme une « *organisation cognitive de l'expérience du locuteur* » (Bybee, 2008). En ce sens, les unités linguistiques représentent des routines cognitives, autrement dit, ce sont « *des patrons récurrents d'activation mentale, qui ne sont pas stockées dans un endroit particulier du système neural* (Bybee, 2003). » (Aurélien Nardy, Céline Dugua, 2016). De ce fait, le jeu LUCIOLE repose sur les théories basées sur l'usage et accorde une place essentielle à l'input langagier pour l'acquisition d'une langue seconde.

Chapitre 4. Les fonctionnalités du jeu LUCIOLE

Il est important de préciser que les fonctionnalités du jeu LUCIOLE figurent dans les pages du wiki créés par l'équipe du projet LUCIOLE et qui servent de cahier des charges afin de permettre aux collaborateurs d'être informés sur l'état d'avancement du projet.

1. *Se connecter à LUCIOLE*

Au lancement du jeu, LUCIOLE propose deux modes de connexions : le mode online et le mode offline.

- **Le mode online** est réservé aux élèves qui participent au protocole expérimental. Un code est alors attribué pour chaque école faisant partie de l'expérimentation Fluence et le compte de chaque élève est enregistré dans l'application. Les traces de chaque joueur sont sauvegardées et stockées dans le logiciel avec un identifiant Fluence correspondant. Par la suite, l'application récupère la liste de tous les comptes des joueurs ayant joué à LUCIOLE.
- **Le mode offline** concerne plutôt les personnes ne faisant pas partie
- de l'expérimentation et souhaitent tester le jeu (en l'occurrence, les personnes qui travaillent sur le projet LUCIOLE). Il n'y a aucun code à choisir pour ce mode. En revanche, il existe un espace de stockage pour ces profils afin de sauvegarder leur progression dans le jeu.

2. *Structure du jeu LUCIOLE*

2.1. *Les missions*

Fondé sur la narration, le jeu LUCIOLE est composé d'un ensemble de missions, organisé en difficulté croissante et en fonction de l'année d'expérimentation. Dans chaque mission, on retrouve différentes thématiques abordées, telles que les couleurs, les nombres, les vêtements, les animaux, des éléments culturels. Voici un tableau qui récapitule les différentes missions pour chaque année expérimentale :

Année 1	Année 2	Année 3
Mission 0	Mission 5	Mission 9
Mission 1	Mission 6	Mission 10
Mission 2	Mission 7	Mission 11
Mission 3	Mission 8	Mission 12 ou mission Q
Mission 4	-	-

Figure 3. Tableau des missions réparties par année d'expérimentation

Il faut remarquer que la particularité de la mission Q ou la mission 12 (telle qu'elle a été programmée dans le jeu) correspond au quadri-chromi-no, un jeu débloqué à la fin de la narration du jeu LUCIOLE et reprend le principe des règles des dominos. L'objectif de la mission Q est de permettre à l'enfant de réviser le lexique et de développer ses compétences en anglais oral indépendamment de ce qu'il a appris pendant les différentes sessions de jeu, et ce en proposant une plus grande quantité de contenu, tout en gardant une structure et un matériel ludique du jeu.

2.2. La structure des missions

Chaque mission est constituée d'une liste d'activités et est structurée suivant l'approche didactique et pédagogique, **PPP** (*Presentation, Practice, Performance*)¹¹. Cette méthode, qui se traduit par les phases **PEC**, *Présentation, Entraînement, Contextualisation*, dans le jeu LUCIOLE, est tout à fait conforme au déroulement d'une séance de langue pour le cycle 2.

- La phase de **Présentation** correspond la plupart du temps à la première étape de la mission. Elle commence le plus souvent par une scénette d'ouverture (ou une narration introductive) et poursuit par une présentation du contenu linguistique.
- Quant à l'étape d'**Entraînement**, elle permet au joueur de travailler un point de la langue, grâce aux différentes activités proposées dans le jeu. Il a également la possibilité de refaire les activités pour s'entraîner.

¹¹ Cette approche pédagogique, *Presentation, Practice, Performance*, est une méthode d'enseignement des langues, visant à organiser les leçons liées à la compétence linguistique suivant la structure PPP. Pour plus de détails, voir la page du wiki : <https://wiki.lezinter.net/ /Presentation, Practice, Performance>

- En ce qui concerne la phase de *Contextualisation*, le contenu des activités peut être plus complexe pour l'enfant, car les objectifs linguistiques compris dans ces activités sont plus nombreux. Cela donne la possibilité à l'enfant de mettre en pratique ce qu'il a acquis dans les deux phases précédentes.

Notons bien que ces trois phases, structurant le déroulement du jeu, ne s'enchaînent pas nécessairement dans un ordre **PEC** pour chaque mission. Néanmoins, on peut dire que la contextualisation joue un rôle décisif dans l'organisation d'une mission.

2.3. Les activités

Chaque phase d'une mission comporte plusieurs types d'activités, appelés aussi des mini-jeux. Ces activités permettent d'atteindre un objectif linguistique précis, tout en conservant un environnement ludique du jeu. Ainsi, LUCIOLE se caractérise par trois types d'activités principales :

- **Recherche visuelle** ou recherche d'objets souvent retrouvée dans la phase de présentation ;
- **Simon** repose sur la mémorisation d'une séquence dictée par la machine ;
- **Question-réponse** composée de trois sous-activités :
 - **QCM** en touchant un item à l'écran ;
 - **Glisser déposer** pour déposer un item dans une zone de dépôt ;
 - **QR code** pour obtenir une chaîne de caractère.

Il faut souligner que l'intérêt d'intégrer une activité en scannant un QR code permet de mettre en relation deux espaces opposés, l'espace physique et l'espace numérique. Le QR code dans LUCIOLE n'est utilisé qu'à partir de l'année 2, donc de la mission 5. L'élève ou le potentiel joueur devra aller chercher l'image du QR code auprès de son enseignant et le scanner comme indiqué dans le jeu. Nous citons un exemple d'utilisation du QR code dans une activité. Lorsque

« [...] le joueur souhaite prendre le train, il devra acheter/valider son billet, pour cela il devra trouver le QR code correspondant à son billet de train, c'est en le validant que l'histoire se déblocquera et que le joueur montera à bord du train.¹² ».

¹² https://wiki.lezinter.net/ /LUCIOLE:Sp%C3%A9cifications/QR_code

La figure ci-dessous illustre le déroulement d'une mission, ainsi que les types d'activités qu'on peut retrouver dans une mission :



Figure 4. Un exemple de la structure d'une mission et de son contenu¹³

2.4. Les modalités de progression dans le jeu

L'équipe du projet LUCIOLE a mis en place quatre types de modalités de progression pour développer la compétence orale de l'anglais chez les enfants du cycle 2 via un jeu sérieux :

- **Map** : le joueur ne peut passer à l'activité suivante qu'après avoir validé l'activité en cours. Dès lors qu'une mission est terminée, une nouvelle carte est déverrouillée. Voici l'exemple d'une carte :

¹³ [https://wiki.lezinter.net/ /Fichier:D%C3%A9coupage_mission_\(LUCIOLE\).png](https://wiki.lezinter.net/ /Fichier:D%C3%A9coupage_mission_(LUCIOLE).png)

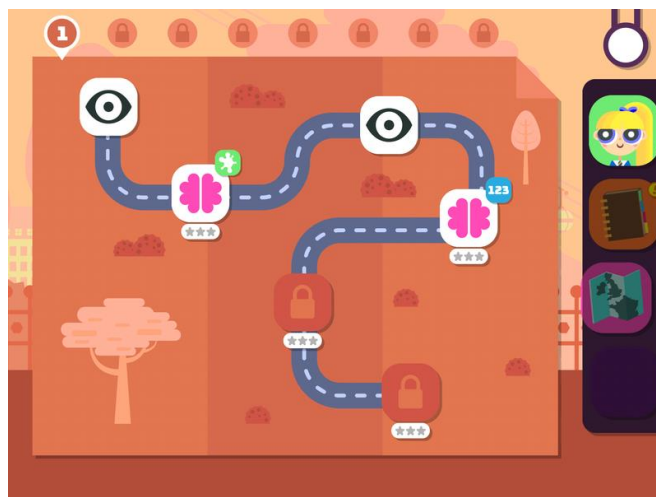


Figure 5. Map d'une mission¹⁴

- **Agrégation du score** : toutes les activités contiennent un certain nombre d'étoiles, allant de 1 jusqu'à 3, hormis les activités qui concernent la phase de présentation, entre autres la recherche visuelle. Cette agrégation du score des activités permet à l'élève de s'autoévaluer et voir sa progression dans le jeu.
- **Training center** : ou le centre d'entraînement permet au joueur de refaire des activités déjà jouées afin d'améliorer son score.
- **Checkpoint** : le mécanisme de checkpoint est opérationnel qu'à partir de l'année 2 dans LUCIOLE et est le plus souvent retrouvé dans la phase de contextualisation. L'idée est de pousser l'élève à rejouer les missions afin d'obtenir un meilleur score. Pour cela, il doit obtenir un nombre d'étoiles suffisant pour passer à la mission suivante.

2.5. Le déroulement des sessions de jeu

Rappelons que le jeu LUCIOLE, tout comme les autres jeux Fluence, a été intégré en classe selon le protocole expérimental que nous avons décrit *supra* (voir la figure 2) et dans lequel il a été précisé qu'une session de jeu dure 20 minutes par séance. Il faut savoir qu'une fois que les 20 minutes se sont écoulées, le système suggère au joueur de quitter la partie, sauf si l'enseignant l'autorise à jouer plus de temps. Une fois que le joueur quitte le jeu, il devra alors reprendre le début de l'activité inachevée lors de la session suivante. La

¹⁴ <https://wiki.lezinter.net/ /LUCIOLE:Sp%C3%A9cifications/Map>

figure ci-dessous montre le déroulement d'une session de jeu, où le joueur devra reprendre l'activité à la sous-mission 2.1 :

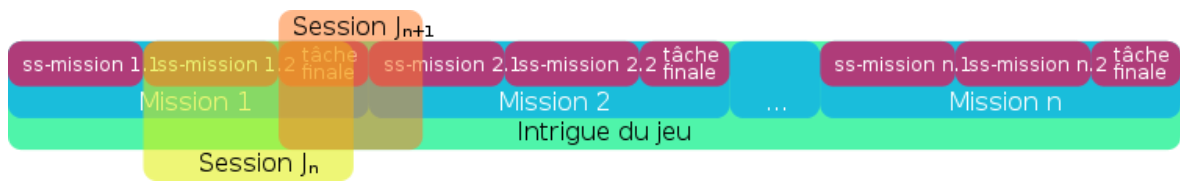


Figure 6. Déroulement d'une session de jeu¹⁵

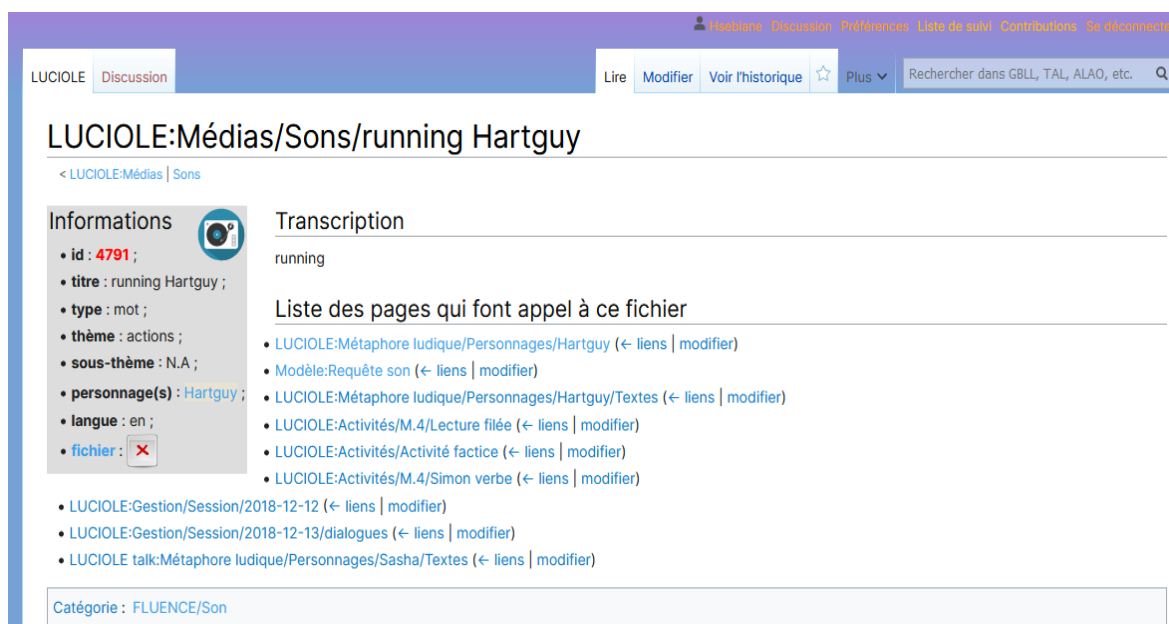
¹⁵ https://wiki.lezinter.net/ /Fichier:Structure_d%27une_mission_du_jour.png

Chapitre 5. Les données collectées

1. Les mediaWiki

Pour la réalisation du jeu LUCIOLE, plus de 1500 sons ont été enregistrés en studio avec des locuteurs natifs, puis transcrits selon les besoins du projet. Pour le stockage des sons, une banque de sons a été alors créée, il en est de même pour les activités.

Pour cela, les développeurs du projet ont utilisé une extension de MediaWiki, *cargo*¹⁶, afin d'avoir une base de connaissances des activités et des sons utilisés pour le jeu pour ensuite les ajouter dans la base de données du projet. C'est ainsi que chaque son et chaque activité peuvent être visualisés depuis une page wiki unique à partir des modèles `{{ref son}}` ou `{{ref activité}}`. On retrouve également une description et un stockage des informations concernant une activité `{{Infos activité}}` ou un son `{{Infos son}}`. Ci-dessous, un mediaWiki qui contient les informations d'un son du jeu :



The screenshot shows a MediaWiki page titled "LUCIOLE:Médias/Sons/running Hartguy". The page layout includes a navigation bar at the top with user options like "Hebbians Discussion Préférences Liste de suivi Contributions Sa déconnecter". Below the navigation bar, there are tabs for "LUCIOLE" and "Discussion", and a search bar. The main content area is divided into several sections: "Informations" (with fields like id: 4791, titre: running Hartguy, type: mot, thème: actions, sous-thème: N.A, personnage(s): Hartguy, langue: en, fichier: [redacted]), "Transcription" (containing the text "running"), "Liste des pages qui font appel à ce fichier" (listing various internal links), and a "Catégorie" field (containing "FLUENCE/Son").

Figure 7. Exemple d'un mediaWiki d'un son

¹⁶ D'après le site www.mediawiki.org, Cargo est une extension de MediaWiki qui fournit une manière allégée d'enregistrer et rechercher les données contenues dans les appels aux modèles, comme les boîtes d'information.

Le même processus a été utilisé pour l'enregistrement et le stockage des images utilisées dans le jeu LUCIOLE, en recourant à l'extension *cargo* avec les modèles [{{Ref image}}](#) et [{{Infos image}}](#).

2. Tests et expérimentations

Parmi les données recueillies dans le cadre du projet LUCIOLE, on trouve :

- les Pré-tests et les post-tests qui ont permis de former des groupes expérimentaux homogènes avec les groupes « *contrôle* » des logiciels EVASION et ELARGIR. Il faut savoir aussi que les post-tests servent de pré-tests pour chaque année d'expérimentation (la figure 8, ci-après récapitule les tests réalisés) ;
- un questionnaire aux parents sur les langues utilisées en famille ;
- un questionnaire aux enseignants sur l'usage de LUCIOLE en classe, les problèmes rencontrés, la motivation des élèves, etc. ;
- les traces d'interaction suite aux tests effectués pour chaque élève faisant partie de l'expérimentation. Cela permet de récupérer les sons écoutés par l'enfant et le contexte dans lequel il a été écouté, le temps d'entraînement et la progression dans le jeu.

Voici un tableau qui récapitule les pré et post-tests réalisés dans le cadre du projet LUCIOLE :

	<i>Année 1</i>	<i>Année 2</i>	<i>Année 3</i>	<i>Etude complémentaire</i>	
Pré-tests	CP : nov/déc 2017	CE1 : mai/juin 2018	CE2 : mai/juin 2019	CE1 et GSM : janv/fév 2021	
	pas de PO (Production Orale) enregistrée, uniquement des images pour provoquer la PO	plusieurs images pour provoquer la PO	PO enregistrée	PO enregistrée	
Post-tests	CP mai/juin 2018	CE1 : mai/juin 2019	CM1 : 2020	CE1 et GSM : mai/juin 2021	
	même exercice, plusieurs images pour provoquer la PO	PO enregistrée	Pandémie COVID-19 (retardement des pot-tests) PO enregistrée	du PO enregistrée	

Figure 8. Tableau récapitulatif des tests réalisés par année d'expérimentation¹⁷

3. Définition des concepts utilisés

Parmi les données collectées pour l'analyse et l'évaluation du jeu LUCIOLE, nous retrouvons les traces d'interaction qui nous ont permis de réaliser ce travail. Nous devons alors poser la définition des concepts que nous utiliserons afin de mieux appréhender l'analyse des données, dans ce qui suit.

3.1. Qu'est-ce qu'une trace dans les EIAH ?

La notion de « trace » est polysémique et est souvent utilisée dans le langage courant pour désigner une empreinte ou une marque. Toutefois, l'émergence des Environnements Informatiques d'Apprentissage Humain (EIAH) amène à repenser la notion de trace pour parler des « traces numériques ». Deux chercheurs au sein du LIRIS de l'équipe Silex¹⁸, Laflaquière Julien et Prié Yannick, ont tenté de concevoir une nouvelle approche

¹⁷ <https://wiki.lezinter.net/ /Exp%C3%A9s:LUCIOLE>

¹⁸ L'équipe Silex (Supporting Interaction and Learning by Experience) est une ancienne équipe de recherche du laboratoire LIRIS et n'est plus active.

des traces numériques dans le domaine des EIAH, afin de proposer une réflexivité aux apprenants/utilisateurs, dans le cadre d'une plateforme d'apprentissage collaboratif en ligne (2007). Ils considèrent la notion de trace numérique comme :

« [...] un enregistrement automatique d'éléments d'interaction entre un utilisateur et son environnement, dans le cadre d'une activité donnée » (Laflaquière et Prié, 2007).

Ces éléments d'interaction, qui constituent une trace numérique, résultent des observations de l'activité d'utilisation d'un Environnement Informatique (EI), appelés aussi « *observés* » (Laflaquière et Prié, 2007). Les observés d'une trace sont représentés par un *modèle de trace* ou « *le vocabulaire de la trace* », tel qu'il a été décrit par Laflaquière et Prié (2007). De même, l'ensemble de ces observés est structuré par leurs relations, dans une séquence temporellement située (Lund et Mille, 2009), on parle alors d'une *trace modélisée*.

3.2. Le principe d'un Système à Base de Traces modélisées (SBTm)

S'appuyant sur l'approche des Systèmes à Base de Traces modélisées (SBTm) pour définir un cadre conceptuel permettant de mieux concevoir les traces numériques, les chercheurs de l'équipe Silex ont élaboré un *framework* qui permet de décrire un SBTm utilisant des traces numériques (Laflaquière et Prié, 2007), représenté dans la figure ci-après :

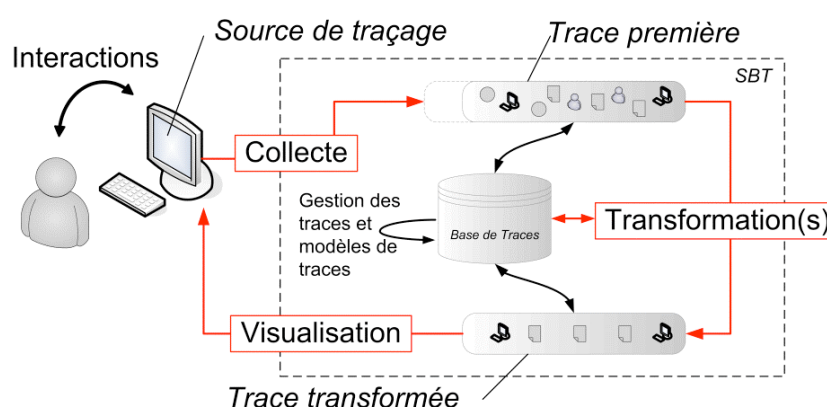


Figure 9. Principe général d'un SBTm (Laflaquière et Prié, 2007)¹⁹

¹⁹ Cette figure est tirée de l'article « Des traces modélisées, un nouveau support pédagogique ? », de Julien Laflaquière, Yannick Prié (2007).

Le SBTm ou le gestionnaire des traces modélisées (Lund et Mille, 2009) permet la manipulation et l'exploitation des traces modélisées et de leurs modèles à partir de la collecte des traces, alimenté par les observés (interactions) entre un utilisateur et un environnement informatique. Il permet des requêtes de transformations à des degrés divers :

- la **collecte** et la **gestion** de la trace première via une source de traçage (interactions des observés entre l'utilisateur et un EI), modélisée selon le premier modèle de collecte et enregistrée dans la base des traces ;
- la **transformation** de la trace première (par sélection, fusion et/ou réécriture) en une trace transformée. La transformation de la trace modélisée permet de définir les indicateurs pour l'exploiter ou réaliser des calculs ;
- la **visualisation** de la trace modélisée à partir de la trace transformée, selon le modèle de traces défini à des fins d'analyse.

3.3. *Qu'est-ce qu'une trace d'interaction dans LUCIOLE ?*

En s'appuyant sur ce que nous avons expliqué dans les points précédents, on considère une trace d'interaction dans LUCIOLE, l'enregistrement d'actions effectuées dans le jeu LUCIOLE par les élèves ayant participé au protocole expérimental. Ces traces d'interaction ont pour but de proposer à l'élève une visualisation de sa progression dans le jeu et des activités jouées, d'un côté ; d'un autre côté, elles permettent aux chercheurs du projet de les collecter pour être exploitées en dehors du jeu, à des fins d'analyse et de validation. C'est cette seconde approche qui nous intéresse dans notre travail.

Nous retiendrons la définition de Pierre-Antoine Champin, selon laquelle une trace est constituée d'« *obsels* » (les éléments observés). Un « *obsel* » est formé sur le même mode qu'un « *pixel* », il permet d'observer un élément de la trace brute pour ensuite la réutiliser – soit en la gardant telle quelle, soit en la transformant – et obtenir des informations (2014). Chaque « *obsel* » comporte : un type, une date, un sujet et d'autres attributs²⁰.

Dans la figure ci-dessous, nous reprenons un exemple extrait du wiki qui montre l'exemple d'une trace constituée de 4 « *obsels* » dans un salon de discussion, dans lequel Alice se connecte à un IRC et envoie un message « *hi world* » et reçoit une réponse de world

²⁰ https://wiki.lezinter.net/Trace_mod%C3%A9lis%C3%A9e

« *hi yourself* ». On observe que les « *obsels* » sont localisés dans le temps (la flèche *t*) et les actions, « envoyer », « recevoir » et « fermer » sont toutes reliées à l'action « ouvrir » du canal IRC :

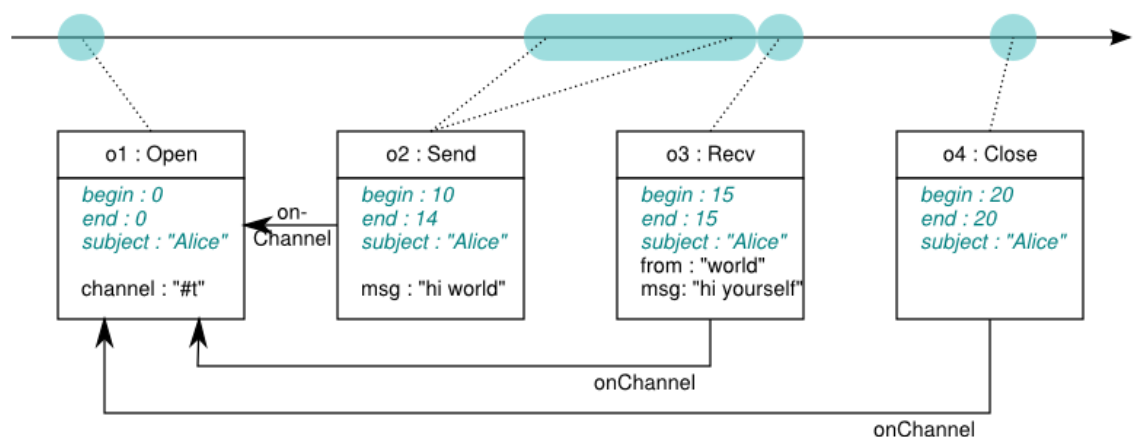


Figure 10. Exemple d'une trace constituée de 4 obsels²¹

Nous reviendrons plus en détail, dans la partie suivante, sur la modélisation des traces LUCIOLE.

3.4. Qu'est-ce qu'un indicateur ?

Une trace est composée d'un ensemble d'observés temporellement situés. Elle comporte toutes les actions effectuées par son utilisateur pendant l'utilisation d'un EIAH. Les observés, constituant les traces numériques, peuvent décrire des informations précises sur les différentes actions de l'utilisateur/apprenant, comme par exemple le temps passé dans une activité, les sons écoutés, les pages consultés, les réponses données, etc. Ces éléments qui « re-tracent » toutes les actions d'un apprenant utilisant un EIAH constituent des indicateurs qui sont définis soit avant, soit après l'apprentissage (Lund et Mille, 2009). D'après Pierre-Antoine Champin, on peut considérer un indicateur comme « *une mesure globale effectuée sur une trace*²² » (2014). Un indicateur « pourrait » former un « *obsel* » particulier à lui-même (Champin, 2014). Chaque indicateur doit nécessairement être daté, afin qu'il puisse être croisé avec d'autres indicateurs.

Cette partie du travail sur les fonctionnalités et les données recueillies pour le projet LUCIOLE est une étape importante, pour mieux appréhender les traces sur lesquelles nous

²¹ https://wiki.lezinter.net/ /Fichier:P.A.Champin-ex_trace.png

²² [https://wiki.lezinter.net/ /Transformation_\(trace\)](https://wiki.lezinter.net/ /Transformation_(trace))

avons travaillé. Elle a permis la prise en main des traces d'interaction issues du serveur afin d'être analysées, *a posteriori*. En effet, la manipulation des traces LUCIOLE que nous expliciterons dans la partie suivante de ce travail, demande de s'imprégner des données pour pouvoir les restructurer et les analyser.

Partie 3

-

Analyse des traces LUCIOLE

Chapitre 7. La prise en main des traces

Dans cette partie du travail, nous allons tenter d'explicitier l'analyse des traces LUCIOLE et les différentes étapes par lesquelles nous sommes passés afin de croiser l'input et les traces d'interaction. Il convient de préciser que le projet sur lequel nous avons travaillé est hébergé sur la plateforme GitLab de la communauté de l'UGA. Outre, les traces et les scripts développés dans le cadre de ce stage, on retrouve également les tickets qui correspondent aux tâches effectuées pendant cette période, selon les besoins du projet.

1. Modélisation des traces LUCIOLE

Pour comprendre la structure des traces sur lesquelles nous avons travaillé tout au long de ces derniers mois, nous allons tenter d'éclaircir les indicateurs et la modélisation des traces LUCIOLE que nous avons analysés.

Pour connaître les sons joués par l'enfant et à quel mot ou énoncé correspondent, il a fallu créer une base de connaissances afin de faire le lien entre les contenus écoutés par les élèves et les contenus correspondant aux activités et à leur progression. C'est cette base de connaissances qui a permis d'évaluer les différents contextes dans lesquels l'élève a écouté tel ou tel son. Pour cela, les développeurs de LUCIOLE ont mis en place un système par « *obsels* » qui utilise les transcriptions des sons et ses attributs afin de les croiser avec les sons écoutés par l'élève.

De ce fait, l'exploitation des traces nécessite la modélisation de celles-ci. Ainsi, un modèle de trace peut définir « les types d' « *obsels* » (type, date, sujet, attributs), les attributs associés à chaque type et les relations possibles entre les types²³» (Champin, 2014). C'est ainsi qu'une société privée, appelée GOTAN, a travaillé sur le jeu LUCIOLE. Elle s'est chargée de la partie développement du projet et plus précisément des informations liées à l'envoi des traces, notamment de la structure de la base de données.

Pour modéliser les traces LUCIOLE, GOTAN a fait la distinction entre deux types d'indicateurs en proposant une modélisation des traces pour chaque type d'indicateur. Ci-

²³ https://wiki.lezinter.net/ /Trace_mod%C3%A9lis%C3%A9

après, la proposition d'une modélisation des traces faite par la société GOTAN selon le type d'indicateur :

- les indicateurs relatifs aux données d'entrée du joueur :

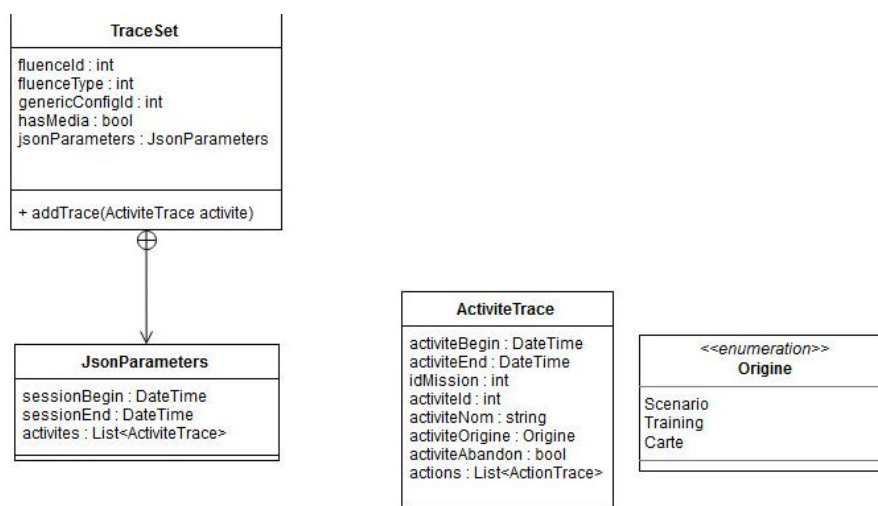


Figure 11. Modélisation des traces de l'input²⁴

La modélisation des traces de l'input concerne le traçage de tous les sons auxquels le joueur a été confronté pendant le jeu. Les traces de l'input contiennent l'attribut et la valeur des indicateurs suivants : *idFluence* (identifiant du joueur) ; *create_date* (la date de création des traces) ; les paramètres json dans lesquels on retrouve les *sessionBegin* et *sessionEnd* (ils indiquent la date et l'heure pour chaque début et fin de session de jeu) et une liste des traces d'activité (*activiteTrace*). Les traces d'activité indiquent la date de début et de fin d'activité (*activiteBegin* et *activiteEnd*), les identifiants de la mission et de l'activité jouée, puis l'origine de l'activité (scenario, training, map), l'abandon de l'activité (qui permet de vérifier par un booléen – vrai ou faux – si l'activité a été jouée), et enfin la liste des traces d'actions (*actionTrace*).

- les indicateurs qui tracent les actions du joueur :

²⁴ https://wiki.lezinter.net/w/images/Mod%C3%A8le_de_trace_LUCIOLE_v1.jpg

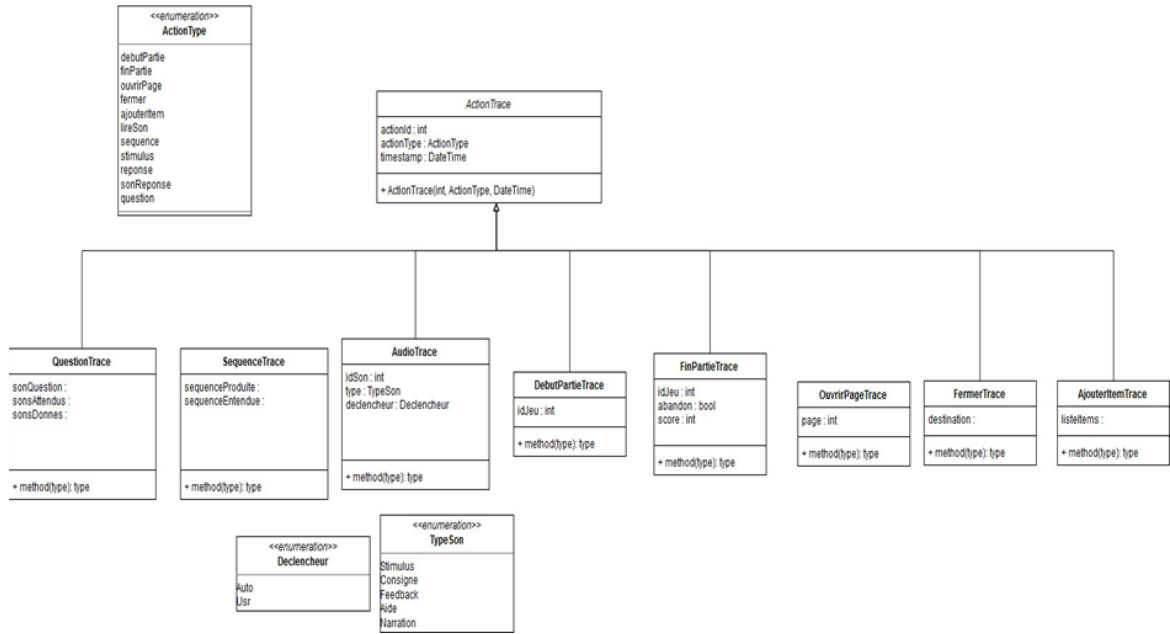


Figure 12. Modélisation des traces d’actions

Les traces d’actions de l’utilisateur concernent la liste des traces d’actions (*actionTrace*). On retrouve les indicateurs d’actions suivants : *idSon* (il permet de récupérer toutes les informations liées au son joué), *typeSon* (désigne le statut du son dans l’activité) ; *actionDeclencheur* (elle indique si le son entendu a été déclenché automatiquement par le système ou par l’utilisateur) ; et enfin, le *timestamp* (l’heure et la date auxquelles le son a été déclenché).

La reconstitution de l’input et la modélisation des traces par « *obsels* », pour enregistrer chaque son écouté par l’élève, étaient nécessaires afin de pouvoir collecter et exporter les traces brutes LUCIOLE.

2. Exportation des traces LUCIOLE

Tout d’abord, dans le cadre de l’analyse des traces LUCIOLE, un nouveau projet a été créé sur GitLab, nommé « *Luciole traces* ». Ce projet est un espace qui nous a permis d’exporter/importer les données et de partager le travail effectué durant cette période. Afin de prendre en main les traces LUCIOLE, nous avons exporté toutes les données hébergées sur GitLab qui constituaient les traces premières collectées (cf. le principe général d’un SBTm, dans la figure 9) ou ce que nous désignons de traces brutes. Cette exportation avait pour but de nous permettre la manipulation et l’exploitation des traces d’interaction pour

être transformées et visualisées. C’est ce que nous expliciterons dans les points suivants. Il convient de noter que l’analyse des traces a été réalisée en Python.

2.1. Les traces brutes

Pour analyser les traces LUCIOLE, il a fallu, dans un premier temps exporter toutes les données brutes qui se trouvent dans un dossier nommé [Traces brutes]. Le dossier [Traces brutes] est constitué de quatre dossiers répartis par année d’expérimentation. Nous tenons à préciser que dans le cadre du projet LUCIOLE, nous avons trois années de tests (2018, 2019 et 2020) et une année d’étude complémentaire (2021). Pour chaque année, on retrouve quatre fichiers XML issus du serveur des traces. Nous schématisons le dossier des traces brutes et les fichiers XML pour chaque année de test dans le diagramme suivant :

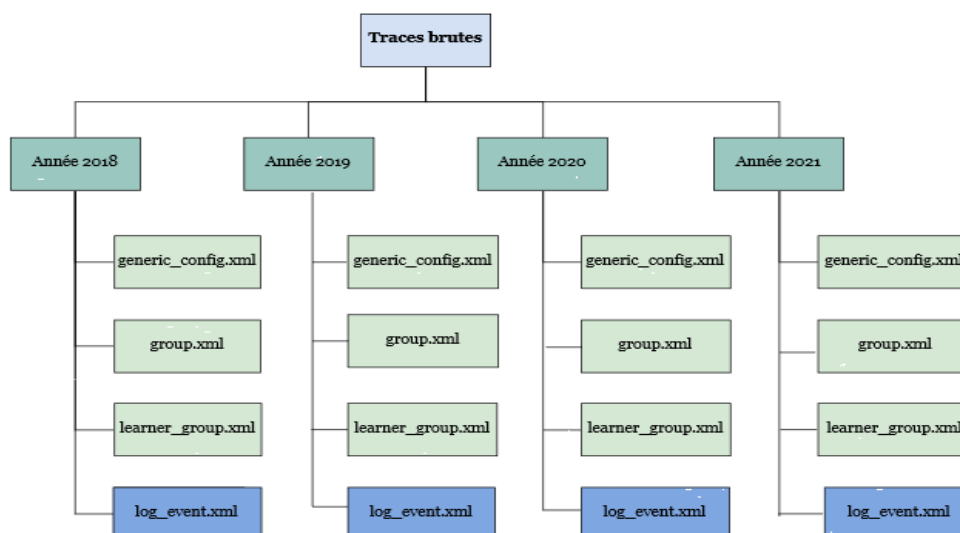


Figure 13. Diagramme des traces brutes

Comme nous pouvons le constater sur le diagramme des traces brutes, nous avons plusieurs fichiers XML. Chaque fichier XML est identifié par des éléments, tels que le nom de l’application, la date d’exportation des traces (annee/mm/jj) et le type de traces. Par exemple, le fichier « *learner_group* » de l’année 2018 est identifié de la façon suivante : {export-luciole-2018-10-07.learner_group.xml}. Toutefois, nous nous sommes intéressées uniquement aux fichiers {log_event.xml}, car ce sont les fichiers qui contiennent toutes les traces des joueurs, à savoir les traces de l’input et d’actions.

En effet, les fichiers XML {export-luciole-annee-mm-jj.log_event.xml} sont très volumineux (de 109791 ko jusqu'à 298662 ko), au vu de la quantité des traces qu'ils comportent. Ils sont scindés en balise <row>, comprenant un identifiant de la session <id>, les traces d'actions <json_parameters>, l'identifiant de l'application <application_context_id> (qui correspond à 3 pour LUCIOLE), mais aussi l'identifiant du joueur <fluence_id> et d'autres balises concernant la date, le groupe ... Ci-après, un exemple des traces brutes au format XML :

```

<row>
  <id>530</id>
  <json_parameters>{"activites": [{"actions": [], "idMission": 0, "activiteId": 3607, "activiteEnd": 1517233550, "activiteType": "Narration",
  <has_media>>false</has_media>
  <create_date>2018-01-29T14:48:36.807</create_date>
  <update_date>2018-01-29T14:48:36.807</update_date>
  <generic_config_id>52</generic_config_id>
  <application_context_id>3</application_context_id>
  <fluence_type>1</fluence_type>
  <fluence_id>201713672</fluence_id>
  <group_id>634</group_id>
</row>

<row>
  <id>537</id>
  <json_parameters>{"activites": [{"actions": [], "idMission": 0, "activiteId": 3607, "activiteEnd": 1517233625, "activiteType": "Narration",
  <has_media>>false</has_media>
  <create_date>2018-01-29T14:56:49.182</create_date>
  <update_date>2018-01-29T14:56:49.182</update_date>
  <generic_config_id>52</generic_config_id>
  <application_context_id>3</application_context_id>
  <fluence_type>1</fluence_type>
  <fluence_id>201713641</fluence_id>
  <group_id>619</group_id>
</row>

```

Figure 14. Traces brutes au format XML

2.2. Découpage des fichiers XML

Afin de faciliter la lecture de ces fichiers XML, nous avons tenté de découper ces fichiers en petits fichiers ouvrables avec un éditeur de texte, dans un script python [*split_file.py*]. Pour cela, nous avons sélectionné une vingtaine de <row> pour chaque fichier XML découpé. Bien que ce découpage des fichiers n'ait pas été nécessairement utilisé dans le travail qui suit, néanmoins cela nous a aidé à définir un formalisme JSON pour la transformation des traces que nous détaillerons dans les points suivants.

3. Formalisme des traces en JSON

Pour définir un formalisme des traces pour chaque utilisateur, nous avons identifié les ID fluence à partir de la balise <fluence_id> et avons sauvegardé uniquement les balises

suivantes : <id>, <json_parameters>, <has_media>, <create_date>, <group_id>. Le choix de ces balises s'est porté sur les besoins du projet pour le calcul des routines. Nous sommes parties du principe que toutes les traces des fichiers XML sont des traces de l'application LUCIOLE. C'est la raison pour laquelle, nous n'avons pas ajouté les balises de <generic_config_id>, <application_context_id> et <fluence_type>, car ce sont des identifiants relatifs au type d'application. En ce qui concerne les dates, nous avons fait le choix de sauvegarder uniquement la date de création <create_date>, puisque celle-ci est toujours similaire à la date figurant dans la balise <update_date>. Ci-dessous, un exemple de fichier de traces pour un seul utilisateur d'après le formalisme défini en JSON :

```
1  {
2    "fluence_id":201714731,
3    "group_id":2214,
4    "session":[
5      {
6        "row_id":74417,
7        "date":"2020-01-10T13:45:07.254",
8        "sessionBegin":1578408778,
9        "sessionEnd":1578408778,
10       "activites":[
11         {
12           "actions":[
13             {
14               "sonId":"5339",
15               "actionId":1,
16               "actionType":"stimulus",
17               "actionTimestamp":1578408779,
18               "actionDeclencheur":"Auto"
19             },
20             {
21               "sonId":"5339",
22               "actionId":1,
23               "actionType":"stimulus",
24               "actionTimestamp":1578408786,
25               "actionDeclencheur":"Auto"
26             }
27           ]
28         },
29         "idMission":5,
30         "activiteId":5297,
31         "activiteEnd":1578408800,
32         "activiteType":"QCM",
33         "activiteBegin":1578408779,
34         "activiteScore":0,
35         "activiteAbandon":true,
36         "activiteOrigine":"Scenario"
37       ],
38     }
```

Figure 15. Formalisme JSON d'un fichier de traces pour un utilisateur

Chapitre 8. Transformation des traces LUCIOLE

1. *Exportation des traces XML en JSON*

Suivant le formalisme JSON défini plus haut, nous avons développé un script en python [*fluence_id.py*] qui permet d'exporter les traces XML en JSON à l'aide d'une bibliothèque *_xmldict_* pour ensuite les structurer d'après le formalisme JSON. Le script prend donc en entrée un fichier XML {*export-luciole-annee-mm-jj.log_event.xml*} et renvoie en sortie un dossier *Fluence_id_annee* contenant les fichiers JSON convertis. Chaque fichier représente un seul utilisateur, identifié par son *fluence_id* [**201712546.json**]. Le fichier agglomère toutes les traces de l'utilisateur/élève. La création d'un fichier JSON pour chaque élève (*Fluence_id*) permettra d'analyser ses traces d'interaction, d'évaluer sa progression dans le jeu et de croiser les résultats de celui-ci avec les résultats obtenus aux posts-tests.

2. *Extraction des traces EVASION et ELARGIR*

Nous avons repéré des traces EVASION et ELARGIR dans les traces brutes en XML. Grâce aux identifiants des indicateurs *<generic_config_id>* et *<application_context_id>*. Nous avons pu les distinguer et les séparer des traces LUCIOLE. Ainsi, l'extraction des traces EVASION et ELARGIR a été traitée dans le script [*fluence_id.py*]. Pour cela, nous avons distingué l'identifiant de l'indicateur [*generic_config_id*] :

- EVASION : [*generic_config_id*]=50
- ELARGIR : [*generic_config_id*]=51
- LUCIOLE : [*generic_config_id*]=52

Les traces EVASION et ELARGIR ont été sauvegardées dans un dossier, classé par année d'expérimentation.

Chapitre 9. Désambiguïsation des traces LUCIOLE

L'agglomération des traces de chaque utilisateur dans un fichier JSON a permis d'identifier des anomalies au niveau des traces. La détection des indicateurs qui posaient problème dans les traces a nécessité de passer en revue toutes les activités de sessions, à partir d'un échantillon de fichiers JSON. Pour ce faire, nous avons procédé en trois étapes :

- Fusion des observés identiques ou mal identifiés ;
- Traitement des actions en faisant appel à un utilitaire ;
- Gestion des sessions.

1. Fusion des indicateurs

Nous avons constaté dans les traces LUCIOLE, des données identiques ou ne correspondant pas à celles qui figurent dans la base de données. Cette situation a nécessité la fusion des traces de façon à soulever ces ambiguïtés. Nous citons ci-après, les indicateurs qui constituaient des anomalies au niveau des traces et la solution apportée :

- [sessionEnd] = [sessionBegin] : les similitudes retrouvées dans le début et la fin de session ne permettent pas de mesurer le temps de session à partir de l'horodatage indiqué.
 - ⇒ Nous avons calculé la longueur de la session [sessionLength] avec le dernier [activiteEnd] de l'activité, puis le soustraire à [SessionBegin].
- [activiteId] < 1000 : lorsque l'identifiant de l'activité est inférieur à 1000, cela signifie que l'activité a été mal identifiée et ne figure pas dans la base de données des traces, car techniquement tous les identifiants des activités sont supérieurs à 1000.
 - ⇒ Pour y remédier, nous avons supprimé [activiteId] < 1000, puis nous avons fusionné les [actions] à l'activité précédente, tout en sauvegardant le dernier [activiteEnd] de l'activité.
- [actionTimestamp] = [actionTimestamp] : nous avons repéré des timestamp identiques de l'indicateur [actionTimestamp]. En effet, il est impossible d'effectuer deux actions en même temps.
 - ⇒ De ce fait, nous avons vérifié que l'identifiant du son lu [sonId] et le timestamp [actionTimestamp] soient similaires aux suivants. Ceci dit, cette

action se réitère deux fois lorsque cette condition est confirmée. Et nous avons gardé uniquement l'action déclenchée par l'utilisateur.

Dans le tableau suivant, nous résumons les problèmes d'horodatage et d'identification retrouvés dans les indicateurs cités plus haut et les solutions apportées ci-contre :

<i>Anomalie</i>	<i>Solution</i>
$[sessionEnd] == [sessionBegin]$	$[sessionLength] = [activiteEnd][-1] - [sessionBegin]$
Si $[activiteId] < 1000$	supprimer, fusionner aux $[actions]$ de $[activiteId]$ précédente, puis sauvegarder $[activiteEnd] [-1]$
Si $[actionTimestamp] == [actionTimestamp]$	vérifier d'abord si $[sonId] + [actionTimestamp] == [sonId] + [actionTimestamp][+1] \rightarrow$ sauvegarder $[actionDeclencheur] == usr$

Figure 16. Fusion des indicateurs

2. Réécriture des traces

Au vu des anomalies retrouvées dans certains indicateurs des traces, nous avons tenté de soulever ces ambiguïtés, en nous appuyant sur un fichier source, *[Scenario.json]*. Le fichier *[Scenario.json]* se trouve dans les traces brutes et contient tous les identifiants que l'on retrouve dans les traces. Les données y sont agencées dans un ordre croissant, en partant des identifiants des missions, puis des activités, et enfin les sons, selon le type de son (stimulus, réponse, consigne, feedback, etc.) dans l'activité.

En nous aidant du fichier *[Scenario.json]*, nous avons procédé, en premier lieu à la préparation des traces. Ensuite, nous avons créé un utilitaire qui utilise le même fichier source pour restructurer les traces.

2.1. Préparation des traces à l'aide d'un fichier source

D'abord, nous avons généré – à partir du fichier source, *[Scenario.json]* – une liste de sons pour la comparer aux sons existants dans les traces, dans un script *[verif_sonid.py]*. Le fichier de sortie *[sons_manquants.json]* contient tous les sons manquants dans le fichier source. De cette façon, nous avons pu vérifier tous les sons qui ne figurent pas dans le

[*Scenario.json*], en nous référant aux mediaWiki, afin de voir la concordance des sons avec ceux situés dans les traces.

Cette étape du travail nous a permis de traiter certains sons manquants que nous avons retrouvés dans la base de connaissances des traces LUCIOLE. Ces sons ont été traités via [*Scenario.json*].

2.2. Traitement des actions dans les traces

Avant de traiter les sons dans les traces, il a fallu créer un utilitaire [*luciole-specs.py*] qui nous permet de restructurer les traces en initialisant une instance avec le fichier [*Scenario.json*]. Cet utilitaire possède trois classes [*Luciole, Mission et Activité*] et sert à lever l'ambiguïté des identifiants des sons non traités dans le [*Scenario.json*] et d'ajouter un attribut [*typeSon*] à chaque action de l'activité. Ce traitement a été effectué en plusieurs étapes :

- **Traitement des ID de sons à trois chiffres** : certains identifiants de sons contiennent trois chiffres. Cela représente une anomalie dans les traces, car tous les identifiants comportent quatre chiffres ;
 - ⇒ à l'aide des documents du mediaWiki et du [*Scenario.json*], nous avons remplacé les ID de sons à trois chiffres par les bons identifiants à quatre chiffres.
- **Traitement des ID de sons compris entre 1000 et 2000** : il s'avérait que les identifiants à cinq chiffres ont été tronqués ;
 - ⇒ nous avons utilisé une fonction *jeu.inferre_5chiffres* se trouvant dans les classes [*Luciole et Activité*] de l'utilitaire.
- **Ajout de l'attribut [*typeSon*] pour chaque action** : nous avons remarqué que l'indicateur [*actionDeclencheur*] est identifié comme un stimulus dans la plupart des actions ;
 - ⇒ nous avons ajouté un nouvel indicateur aux actions des traces, [*typeSon*]. De cette manière, chaque action possède un attribut [*typeSon*] qui précise le type d'action d'après [*Scenario.json*], en utilisant la fonction *jeu.getActivite(idAct).type_son(idSon)* de l'utilitaire [*luciole-specs.py*].

Nous résumons le traitement des actions dans les traces dans le tableau ci-dessous :

<i>Action</i>	<i>Traitement</i>
- Traitement des ID de sons à trois ;	Usage du mediaWiki + [Scenario.json] → action.get("sonId")
- Traitement des ID de sons compris entre 1000 et 2000 ;	Appel de [luciole-specs.py] → jeu.infere_5chiffres
- Ajout de l'attribut [typeSon] pour chaque action ;	Appel de [luciole-specs.py] → jeu.getActivite(idAct).type_son(idSon)

Figure 17. Traitement des actions dans les traces

La désambiguïsation des traces LUCIOLE a fait l'objet d'un fichier python [nettoyage.py], dans lequel toutes les étapes de la réécriture et la restructuration des traces y sont englobées. Ce script python est appelé dans un autre script de départ [fluence_id] à partir d'une fonction `cleanup_json` afin d'optimiser les différents traitements des traces. De même, nous avons gardé une trace des modifications effectuées sur les traces LUCIOLE à l'aide d'un module de journalisation (*logging* en anglais). Nous avons fait appel à l'outil de logging via le script [mylog.py] dans les codes développés pour la transformation et la désambiguïsation des traces, de sorte que les modifications apportées soient conservées.

3. *Gestion des sessions de jeu*

La réécriture des traces nous a permis d'observer des sessions de jeu très courtes et qui ne durent qu'environ 20 secondes. Or, on considère qu'une session jouée dure *a minima* 10 minutes. Ceci étant dit, lorsqu'on retrouve un écart entre deux sessions de jeu de moins de 10 minutes, celles-ci ne peuvent pas être considérées comme deux sessions différentes. Notre hypothèse émanant de cette observation, est que le joueur lance une session sans faire les activités, puis il relance une seconde session de jeu. Toutefois, du point de vue de l'expérimentation, l'élève n'est censé jouer qu'une seule session par jour/séance. C'est la raison pour laquelle, on ne tient compte que d'une seule session jouée le même jour et pendant une durée de 10 minutes ou plus.

De plus, nous avons également remarqué dans les fichiers JSON, que les sessions de jeu ne sont pas toujours structurées dans l'ordre dans lequel elles ont été jouées. Par conséquent, ces doublons de sessions dans les traces ne nous permettent pas de faire le recensement des indicateurs globaux pour le nombre des sessions de jeu et le temps de session pour chaque utilisateur.

De ce fait, nous avons trié les sessions en comparant tous les éléments qui constituent une session (c'est-à-dire, les actions, les activités et la session) à la session suivante, dans un script [*session_tools.py*]. Ainsi, la gestion des sessions de jeu a été faite en deux étapes :

- la suppression des doublons de sessions : nous avons supprimé le doublon de sessions, lorsque tous les éléments des sessions (s1) et (s2) sont identiques.
- la fusion des sessions espacées de moins de 10 minutes : nous avons fusionné les sessions (s1) à (s2) en ajoutant un champ à la session, nommé [interruptions] pour le temps d'inactivité de session. Ce champ d'interruptions nous permet de connaître le temps d'interruption de session, en stockant l'identifiant de la session ["id"], le début de session ["begin"], la fin de session ["end"] et la longueur de session ["length"].

Chapitre 10. Calcul des indicateurs

Un deuxième niveau de transformation des traces a consisté à définir les différents indicateurs qui permettent de calculer la progression de chaque utilisateur pour chaque année d'expérimentation. Pour ce faire, il a fallu créer des routines pour le calcul de ces indicateurs. Ces routines permettront de confronter les pré et post-tests aux effets du système (les traces collectées) afin d'apporter des éléments de réponses aux questions posées par l'équipe pédagogique, comme l'horodatage, le score obtenu, les nombre d'activités jouées par session, l'usage du carnet de notes... Cette tâche a été divisée en trois parties, selon la catégorie des indicateurs recensés : les traces des autoévaluations, les indicateurs globaux et les indicateurs du carnet de notes.

1. *Le constructeur CSV*

Afin de générer des fichiers CSV pour le calcul des indicateurs, il a fallu faire un constructeur pouvant créer un objet et stocker les informations pour les générer dans un fichier sous format CSV pour chaque année de test. Le fichier CSV a été construit de façon à pouvoir le manipuler avec un autre programme, le langage R. Pour cela, chaque fichier CSV prend une ligne pour chaque utilisateur, identifié par son ID ["fluence_id"] en stockant chaque valeur de l'indicateur dans une colonne. Voici le prototype que nous avons suivi pour générer les fichiers CSV :

```
{"fluence_id":1234567,"groupeColonnes1":{"nom_colonne":"valeur","nom_colonne":"valeur"}, "groupeColonnes2":{"nomcol":"val"}}
```

En revanche, nous nous sommes retrouvés avec plusieurs colonnes afin de pouvoir les croiser par la suite, les unes avec les autres, à l'aide du langage R²⁵.

2. *Création des routines pour le calcul des indicateurs*

Le calcul des indicateurs permettra de faire une corrélation entre l'analyse des traces d'interaction et les pré et post-tests. La création des routines pour le calcul des indicateurs a été réalisée d'après le questionnement de l'équipe pédagogique du projet LUCIOLE. Dans les grandes lignes, les questions posées concernent :

²⁵ Il est important de préciser que cette tâche (c'est-à-dire, le travail avec le langage R) sera réalisée par le statisticien du projet Fluence e-FRAN pour la suite de la recherche.

- le temps de jeu ;
- le nombre d'activités jouées ;
- la progression des joueurs dans le jeu (ou le score obtenu) ;
- l'autoévaluation du joueur ;
- le recours au carnet de notes.

A cet effet, les indicateurs globaux et ceux du carnet de notes ont été générés dans le même fichier CSV, tandis que les indicateurs en lien avec les traces des autoévaluations ont fait l'objet d'un autre fichier CSV, et utilisent le même prototype du constructeur CSV.

2.1. Les traces des autoévaluations

Il convient de noter que pour le calcul des traces des autoévaluations, il a été question de vérifier les occurrences des ID de sons de réponses aux activités d'autoévaluations. Chaque ligne correspond à un utilisateur en spécifiant son ID ["fluence_id"] et prend une colonne par question d'activité, avec pour titre {id_activité#nbfoisrépondu.id_question}. Par ailleurs, nous avons observé une erreur de détection des sons/réponses dans les traces des autoévaluations. C'est pourquoi, nous ne sommes pas allés plus loin pour recenser d'autres indicateurs liés aux traces des autoévaluations.

2.2. Les indicateurs globaux

Les indicateurs globaux permettent de mieux concevoir la progression des joueurs dans le jeu, car ils tiennent compte des paramètres tels que le temps, le nombre d'activités jouées et le score obtenu. Par ailleurs, ils sont organisés en trois parties, du général (les sessions) au particulier (les scores), en passant par chaque activité individuelle. En effet, les indicateurs globaux sont plus ou moins nombreux contrairement aux traces des autoévaluations, y compris en termes de nombre de colonnes. Nous les citons dans le tableau ci-après :

INDICATEURS GLOBAUX	
Session	<ul style="list-style-type: none"> - Temps de jeu : <ul style="list-style-type: none"> o Total o Moyen - Nb sessions de jeu ; - Nb d'activités jouées par session ;
Activité	<ul style="list-style-type: none"> - Pour chaque activité : <ul style="list-style-type: none"> o ID de la première activité jouée dans l'année de test ; o ID de la dernière activité jouée dans l'année de test ; o ID de l'activité la plus jouée pendant cette année ; o ID de l'activité la moins jouée pendant cette année ; o ID de l'activité la plus basse (dans la progression) jouée pendant cette année ; o ID de l'activité la plus avancée (dans la progression) jouée pendant cette année ; - Taux des activités jouées plusieurs fois ; - Nb d'inactivités.
Score	<ul style="list-style-type: none"> - Pour chaque activité : <ul style="list-style-type: none"> o Score minimum obtenu ; o Score maximum obtenu ; o Score moyen obtenu ; - Nb d'étoiles total par session ; - Nb d'étoiles total par activité.

Figure 18. Tableau des indicateurs globaux

2.3. L'usage du carnet de notes

Il faut savoir que l'usage du carnet de notes permet au joueur de retrouver sa progression dans le jeu, mais aussi de demander de l'aide lorsqu'il se trouve en difficulté dans une activité quelconque du jeu. Ainsi, en tenant compte de la clé ["page"] dans les traces JSON, qui fait référence à l'usage du carnet de notes, nous avons pu identifier les ID de sons lancés pour calculer les indicateurs définis.

Le tableau suivant représente les indicateurs en lien avec l'usage du carnet de notes :

USAGE DU CARNET DE NOTES
<ul style="list-style-type: none"> - Nb de sons du carnet de notes écoutés ; - Nb de lancements du carnet de notes ; - Nb de sons écoutés par lancement.

Figure 19. Tableau des indicateurs en lien avec l'usage du carnet de notes

3. *Récapitulatif du travail réalisé et les difficultés rencontrées*

Plusieurs tâches nous ont été confiées durant cette période de stage, toutefois nous avons adapté notre travail en fonction des besoins étudiés pour l'analyse des traces LUCIOLE. En effet, les tickets sur GitLab, qui correspondent aux tâches réalisées, ont été construits au fur et à mesure de l'avancement du projet.

D'abord, l'appréhension de la conception du jeu LUCIOLE a été une étape importante pour la réalisation de notre travail, ainsi que pour la prise en main des traces collectées à partir du serveur, car plusieurs équipes constituées de chercheurs en didactique, de développeurs et d'autres, ont travaillé pendant les trois années précédentes sur le projet et la conception du jeu LUCIOLE. Bien qu'un cahier des charges soit disponible via les pages du Wiki et qui permette de consulter toutes les étapes du travail effectué antérieurement, il n'est pas toujours évident de se situer par rapport aux différentes disciplines (qui sont essentiellement, la didactique, l'EIAH) qui constituent ce projet.

Dès lors que nous avons pris en main les données LUCIOLE et créé un fichier JSON pour chaque utilisateur et pour chaque année de test, des anomalies ont été observées au niveau des traces et ont fait l'objet d'une grande partie de « nettoyage » afin d'y remédier. Il convient de préciser que cette étape de nettoyage et/ou désambiguïsation des traces a été la plus importante en termes de temps, car c'est une étape primordiale pour calculer les indicateurs et par conséquent, faire l'analyse des traces pour la validation du jeu LUCIOLE.

Néanmoins, il a fallu revenir régulièrement sur la partie de nettoyage des traces, car les fichiers JSON sont volumineux et très nombreux à gérer. C'est ainsi que nous avons d'abord généré des fichiers CSV pour le calcul de quelques indicateurs globaux qui nous ont permis par la suite, de détecter des problèmes au niveau des timescodes ou encore des doublons de sessions, par exemple. De même, ce sont les premiers résultats des indicateurs recensés qui ont permis de voir des traces d'autres logiciels que LUCIOLE.

3.1. *Les contraintes rencontrées*

Nous avons été confrontés parfois à des difficultés pour le développement des scripts en python. N'étant pas issue d'une formation en informatique, la prise en main des différents utilitaires utilisés pour le traitement des traces ou pour la création des fichiers CSV avec plusieurs colonnes et manipulables en langage R, n'a pas été une tâche facile, mais

l'encadrement de notre référent de stage nous a été d'une grande aide et nous a permis d'élaborer ce travail.

En raison du temps imparti, certaines tâches n'ont pas pu être réalisées. Nous pensons notamment aux activités du quadrichromino que nous n'avons pas pu traiter. Quant aux routines définies, les indicateurs en lien avec le centre d'entraînement n'ont pas été calculés. Ces indicateurs consistaient à calculer le nombre de lancements du centre d'entraînement, le nombre d'activités jouées depuis le centre d'entraînement et enfin, le nombre de lancements après un checkpoint.

Nous avons également constaté un problème de détection d'une activité interrompue lors des sessions de jeu. Il arrive parfois que l'élève lance une activité, puis l'abandonne. Cette action devrait être détectée grâce à l'indicateur [[“activiteAbandon“] = Vrai], cependant cette action est définie comme [[“activiteAbandon“] = Faux], même lorsque l'activité a été abandonnée. Nous avons pu déterminer un mécanisme de vérification des activités abandonnées par le joueur en réétudiant les traces JSON. Lorsque nous retrouvons plus de trois actions déclenchées²⁶ pour une seule activité, cela signifie que l'activité a bien été jouée. Dans le cas inverse, l'activité est considérée comme une activité abandonnée.

Enfin, il serait intéressant pour nous de voir les résultats qu'a donné le jeu LUCIOLE en croisant les résultats obtenus aux pré et post-tests avec les traces sur lesquelles nous avons travaillé durant ces six derniers mois.

4. Perspectives de l'analyse des traces LUCIOLE et extensions possibles

Ayant pu constater un effet de LUCIOLE grâce aux premiers indicateurs élaborés lors de ce stage, il est important de continuer à corriger les problèmes que nous avons pu repérer dans les traces.

Dans un premier temps, il faudrait mettre en œuvre le mécanisme de vérification des activités abandonnées et revoir, à cet effet, les erreurs liées aux scores d'activités. Au vu

²⁶ Précisons qu'à partir du moment où un ID de son est écouté par le joueur, il est alors considéré comme une action déclenchée.

des nombreux problèmes liés à la gestion des sessions de jeu²⁷, nous estimons qu'il serait intéressant de trouver une méthode qui permette de désactiver automatiquement une session au bout de 20 minutes de jeu, à partir du code source, et suivant le protocole expérimental défini au départ.

Nous avons également observé un temps d'inactivité pendant les activités jouées. Il est vrai que ce problème a été résolu en calculant le temps d'inactivité dans une seule activité jouée. En revanche, le manque de concentration est courant chez l'enfant, le moindre bruit peut le détourner de ce qu'il fait et trouve des difficultés à suivre une activité jusqu'au bout. D'où ce temps d'inactivité dans le jeu. Il serait captivant d'utiliser une vibration dans le jeu LUCIOLE pour alerter le joueur de son inactivité au bout de 3 minutes. La création d'une séquence de vibration pourrait être développée dans Unity pour une extension possible du jeu LUCIOLE.

Dans cette optique, nous avons observé des résultats différents quant au nombre et au temps de sessions de jeu. Selon nous, il est essentiel de voir comment le protocole expérimental a été appliqué en classe. Pour cela, il faudrait prendre en compte l'indicateur de l'ID du groupe en rassemblant tous les fichiers JSON pour chaque utilisateur par ID de groupe. Ainsi, nous pourrions connaître les disparités dans les différents groupes ayant participé aux tests par rapport au protocole expérimental défini. De cette manière, l'indicateur de l'ID du groupe permettrait de voir la progression des élèves d'un même groupe dans le jeu en tenant compte des variables sociolinguistiques.

Une autre question n'a pas été évoquée, c'est celle du type d'activité (recherche visuelle, simon, QCM, glisser déposer). En effet, il serait intéressant de regarder à partir des traces, le type d'activité la moins réussie dans le jeu en croisant cet indicateur avec celui des scores obtenus pour chaque type d'activité. Cela permettrait de faire le lien entre le type d'activité ayant obtenu le score le plus bas avec le sentiment de difficulté des élèves.

Désormais, l'intérêt d'analyser les traces LUCIOLE étant, d'ores et déjà de croiser les effets du système aux pré et post-test. Toujours est-il que l'analyse des traces est en cours. Ce travail va permettre aux chercheurs permanents (cités *supra*) travaillant sur le logiciel

²⁷ Il arrive que le joueur ne se déconnecte pas à la fin d'une session de jeu. La session peut alors rester ouverte pendant plusieurs jours et peut être reprise à partir de la dernière activité jouée. Ce bug peut créer des erreurs de calcul du temps joué, ainsi que du nombre de sessions jouées.

LUCIOLE, de croiser les traces des autoévaluations avec les scores aux pré et post-tests, ainsi que les scores obtenus dans le jeu avec les scores obtenus dans les post-tests. Une analyse des scores obtenus dans le jeu avec les traces des autoévaluations sera également effectuée.

Conclusion et bilan personnel

En conclusion, l'objectif de notre travail était d'analyser les traces LUCIOLE afin de tenter de croiser les analyses des pré et post-tests aux effets du système. Les indicateurs calculés durant cette période de stage peuvent d'ores et déjà permettre à l'équipe permanente du projet LUCIOLE de répondre aux premières questions posées. Désormais, cette analyse permettra d'étudier la progression des élèves dans le jeu, mais aussi de mesurer les effets du système et son apport dans le développement des compétences de la compréhension orale en anglais, en corrélation avec les post-tests réalisés pendant les trois années d'expérimentation.

Bien que nous n'ayons pas pu calculer tous les indicateurs liés aux traces d'interaction, la désambiguïsation et le nettoyage de ces traces, qui a fait l'objet d'une partie importante de ce travail, est non négligeable. Ils permettront un gain de temps précieux pour d'autres calculs d'indicateurs.

Vraisemblablement, les premiers résultats sont encourageants et laissent entendre que le jeu LUCIOLE a eu un effet significatif sur l'apprentissage de l'anglais oral. Il pourrait permettre aux jeunes apprenants de mener un apprentissage autonome à l'aide d'un jeu sérieux.

Bilan personnel

Ce stage de fin d'études fut pour moi très formateur et est un réel atout pour la suite. Même si j'ai pu réaliser plusieurs projets au cours de ce master en Industrie de la Langue, qui portaient essentiellement sur des langages de programmation suivant des sujets de TAL, le fait de me retrouver avec une masse de données à gérer et à restructurer m'a permis de développer mes compétences en programmation, et plus particulièrement en python.

Ma participation au projet LUCIOLE m'a permis de voir les différentes contraintes auxquelles peut être confronté un travail de recherche, surtout la partie technique du travail. Les conditions exceptionnelles liées à la crise du COVID-19 n'a pas toujours été d'une grande aide. Malgré les doutes que j'ai pu avoir par moments concernant mes capacités ou encore à tenter de faire le lien entre le TAL et le travail que j'étais en train d'effectuer, j'ai pu surmonter ces difficultés grâce aux échanges que j'ai eus avec mon encadrant de stage, qui m'a notamment permis de mieux appréhender des nouveaux outils pour la programmation. De plus, les échanges et le partage d'idées pendant les réunions d'équipe

m'a aidé à me repositionner par rapport au travail que je faisais pour faire le lien avec ma formation en TAL.

Il est vrai que j'aurais souhaité aller plus loin dans le projet LUCIOLE, ou du moins pouvoir croiser les premiers résultats des traces avec les pré et post-tests. Quoi qu'il en soit, je retire de cette expérience beaucoup de positif pour la suite.

Bibliographie

- Brougère, G. (2012). Le jeu peut-il être sérieux ? Revisiter Jouer/Apprendre en temps de serious game, *Australian Journal of French Studies*, 49.1, p. 117-129, <https://doi.org/10.3828/AJFS.2012.10>.
- Bybee, J. (2008). *Usage-Based Grammar and Second Language Acquisition*, Handbook of Cognitive Linguistics and Second Language Acquisition, édité par Robinson, P. & Ellis, N. C., Routledge, p. 216-236.
- European Commission (2012). First European Survey on Language Competence (Survey Final report; Education and Training), *Publications Office of the European Union*, http://ec.europa.eu/dgs/education_culture/repository/languages/policy/strategic-framework/documents/language-survey-final-report_en.pdf.
- Laflaquière, J. et Prié, Y. (2007). Des traces modélisées, un nouveau support pédagogique ?, *I2LOR, Environnements cognitifs centrés utilisateur : de la théorie à la pratique*, Montréal, <https://liris.cnrs.fr/Documents/Liris-3708.pdf>.
- Lund, K. et Mille, A. (2009). Traces, traces d'interactions, traces d'apprentissages : définitions, modèles informatiques, structurations, traitements et usages. *Analyse de traces et personnalisation des environnements informatiques pour l'apprentissage humain*, Hermes Sciences Publications, p.21-66, <https://liris.cnrs.fr/Documents/Liris-3967.pdf>.
- Mandin, S. et al. (2021). Expérimentation à grande échelle d'applications pour tablettes pour favoriser l'apprentissage de la lecture et de l'anglais oral, *10e Conférence sur les Environnements Informatiques pour l'Apprentissage Humain*, édité par Michel, Ch., Fribourg / Virtual, p. 118-129, <https://hal.archives-ouvertes.fr/hal03292798/document>.
- Mandin, S. et al. (2021). EVASION, ELARGIR et LUCIOLE : 3 jeux tablettes du projet FLUENCE pour prévenir les difficultés d'apprentissage de la lecture et de l'anglais, *Perspectives de Recherches sur les Usages du Numérique dans l'Éducation*, 2020, Paris (virtuel), <https://hal.univ-grenoble-alpes.fr/hal03187547/document>.
- Nardy, A. & Dugua, C. (2016). Le rôle de l'usage sur le développement des constructions nominales chez les enfants pré-lecteurs, *Travaux de linguistique, Revue internationale de linguistique française*, De Boeck Supérieur, p. 129-148, <https://www.cairn.info/revue-travaux-de-linguistique-2011-1-page-129.htm>.
- VanPatten, B. (2002). Processing Instruction : An Update, *Language Learning* 52:4, University of Illinois at Chicago, p. 755-803.

Sitographie

- <https://liris.cnrs.fr/liris/> [consulté le 27/07/2021]
- <https://www.gouvernement.fr/efran-les-22-laureats/> [consulté le 27/07/2021]
- <https://fluence.prod.lamp.cnrs.fr/> [consulté le 27/07/2021]
- <https://fluence.prod.lamp.cnrs.fr/luciole/> [consulté le 27/07/2021]
- https://wiki.lezinter.net/_/LUCIOLE:Accueil [consulté le 27/07/2021]
- https://wiki.lezinter.net/_/Jeu_d%27action [consulté le 10/08/2021]
- https://wiki.lezinter.net/w/images/775-m%C3%A9moire_M1_DILIPEM-Adrea_Soh.pdf [consulté le 10/08/2021]
- https://wiki.lezinter.net/_/Presentation,_Practice,_Performance [consulté le 10/08/2021]
- <https://perso.liris.cnrs.fr/pierre-antoine.champin/2014/atelier-traces-et-jeux/#14> [consulté le 10/10/2021]

Références des illustrations tirées du Wiki

- https://wiki.lezinter.net/_/Exp%C3%A9s:LUCIOLE/Plan_d%27exp%C3%A9rimentation [consulté le 10/08/2021]
- https://wiki.lezinter.net/_/LUCIOLE:Sp%C3%A9cifications/QR_code [consulté le 22/08/2021]
- [https://wiki.lezinter.net/_/Fichier:D%C3%A9coupage_mission_\(LUCIOLE\).png](https://wiki.lezinter.net/_/Fichier:D%C3%A9coupage_mission_(LUCIOLE).png) [consulté le 22/08/2021]
- https://wiki.lezinter.net/_/LUCIOLE:Sp%C3%A9cifications/Map [consulté le 22/08/2021]
- https://wiki.lezinter.net/_/Fichier:Structure_d%27une_mission_du_jour.png [consulté le 22/08/2021]
- https://wiki.lezinter.net/_/Fichier:P.A.Champin-ex_trace.png [consulté le 02/09/2021]
- https://wiki.lezinter.net/w/images/Mod%C3%A8le_de_trace_LUCIOLE_v1.jpg [consulté le 02/09/2021]

Sigles et abréviations utilisés

CLAPI : Corpus de LAngue Parlée en Interaction

CNRS : Centre National de la Recherche Scientifique

CSV : Comma-separated values

EIAH : Environnement Informatique d'Apprentissage Humain

GIPSA-Lab : Grenoble Images Parole Signal Automatique

IA : Intelligence Artificielle

ICAR : Laboratoire Interactions, Corpus, Apprentissages, Représentations

ID : Identifiant

IRC : Internet Relay Chat

JSON : JavaScript Object Notation

LIDILEM : Laboratoire de Linguistique et Didactique des Langues Etrangères et Maternelles

LIRIS : Laboratoire d'InfoRmatique en Image et Systèmes d'information

LPNC : Laboratoire de Psychologie et NeuroCognition

LUCIOLE : LUdique au service de la CompréhensIon Orale en Langue Étrangère.

PEC : Présentation, Entraînement, Contextualisation

PPP : Presentation, Practice, Performance

SBTm : Système à Base de Traces modélisées

SILEX : Supporting Interaction and Learning by Experience

UGA : Université Grenoble Alpes

XML : Extensible Markup Language

Table des illustrations

Figure 1. Les équipes de recherche par pôle de compétence.....	10
Figure 2. Plan d'expérimentation du projet Fluence	15
Figure 3. Tableau des missions réparties par année d'expérimentation	21
Figure 4. Un exemple de la structure d'une mission et de son contenu	23
Figure 5. Map d'une mission.....	24
Figure 6. Déroulement d'une session de jeu.....	25
Figure 7. Exemple d'un mediaWiki d'un son	26
Figure 8. Tableau récapitulatif des tests réalisés par année d'expérimentation	28
Figure 9. Principe général d'un SBTm (Laflaquière et Prié, 2007)	29
Figure 10. Exemple d'une trace constituée de 4 obsels.....	31
Figure 11. Modélisation des traces de l'input	35
Figure 12. Modélisation des traces d'actions	36
Figure 13. Diagramme des traces brutes	37
Figure 14. Traces brutes au format XML.....	38
Figure 15. Formalisme JSON d'un fichier de traces pour un utilisateur.....	39
Figure 16. Fusion des indicateurs	42
Figure 17. Traitement des actions dans les traces	44
Figure 18. Tableau des indicateurs globaux	48
Figure 19. Tableau des indicateurs en lien avec l'usage du carnet de notes	48

Table des annexes

Annexe 1 Documentation des scripts	60
Annexe 2 Documentation pour les activités abandonnées	63

Annexe 1

Documentation des scripts

☰ README.md 12.6 KB

</> Edit Web IDE Replace Delete

Luciole traces

Routines pour l'extraction des traces Luciole (à lier à terme avec [Luciole stats](#)).
Projet en python

Chaîne de traitements

On part des données brutes situées dans le dossier `Traces brutes`, puis on effectue les traitements suivants :

1. `fluence_id.py` → créé un fichier JSON par utilisateur à partir des traces brutes XML et y agglomère toutes ses traces.
2. `nettoyage.py` → permet de faire les traitements des timescodes, des sons manquants et d'attribuer un type son (stimuli, feedback, reponses, consignes)
3. `session_tools.py` → gère les sessions de jeu (doublons, inactivités) à partir des traces JSON
4. `progression.py` → utilitaire pour créer une barre de progression dans le jeu. Il est appelé dans `fluence_id.py` et `indic_glob.py`
5. `indic_glob.py` → calcule des routines d'après les indicateurs globaux et les indicateurs du carnet de notes et génère les fichiers CSV
6. `autoeval.py` → vérifie les traces des autoévaluations à partir des traces JSON et calcule les autoevals pour générer des fichiers CSV

Utilitaires

1. `luciole_specs.py` → utilitaire qui permet de manipuler les ID de sons, des activités et des missions à partir du fichier `Scenario.json`. Il est appelé dans `nettoyage.py`
2. `csv_builder.py` → utilitaire pour générer des fichiers csv en plusieurs colonnes. Il est appelé dans `indic_glob.py`
3. `mylog.py` → outil de logging qui sert à garder une trace des modifications apportées dans les traces JSON

Autres fichiers

1. `scenario.py` → regarde les doublons des ID de sons à partir du fichier `Scenario.json`
2. `verif_sonid.py` → compare les ID de sons `Scenario.json` avec ceux des traces JSON

Fichiers inutilisés

- `split_file.py` n'est pas nécessairement utilisé, mais peut servir à découper les fichiers .xml en fichier ouvrables avec un éditeur de texte

1. XML → JSON

🔗 `fluence_id.py`

La plupart des scripts développés dans ce projet sont appelés à partir de ce script, `fluence_id.py`. Il permet de générer tous les traitements effectués

Données d'entrée

Le script travaille sur des fichiers XML issus du serveur de traces.

Données de sortie

Les fichiers résultants du script sont des fichiers convertis en JSON. Chaque fichier JSON représente un utilisateur (identifié par son `fluence_id`). Le fichier agglomère toutes les traces de l'utilisateur.

Dossiers par défaut

Les données d'entrée se trouvent par défaut dans le dossier `Traces brutes` dans un dossier contenant l'année et le fichier s'appelle comme le dossier le contenant avec le suffixe `export-luciole-[année-mm-jj]-log_event.xml`

Les données de sortie sont réparties entre plusieurs dossiers :

- Traces `evasion+elargir/[année]/Evasion` (ou `Elargir`) pour les traces issues des autres logiciels Fluence
- Traces `transformées/Fluence_id_[année]` pour les traces des utilisateurs

Nettoyage des traces

- Les traces `Evasion` et `Élargir` sont extraites des données :
- Nettoyage en appelant : `nettoyage.py`
 - correction de certains `id` de sons
 - correction des `id` de sons compris entre 1000 et 2000
 - correction des `timescodes` identiques
 - correction de certains `id` d'activités < 1000
 - correction de certains `typeSon` et `activiteType`
 - gestion de l'inactivité :
 - Quand le temps entre 2 sessions est inférieur à `max-inactivity` (10 minutes par défaut), on fusionne les sessions, la session obtenue a pour durée la somme des durées des deux sessions, mais les temps d'interruption sont listés (et décomptés du temps de jeu effectif)
 - Quand l'écart entre 2 actions est supérieur à `min-inactivity` (3 minutes 30 par défaut), on comptabilise ce temps comme de l'inactivité pendant le jeu (cf. calcul du temps de jeu effectif) mais la durée de la session n'est pas affectée
 - sauf si l'écart entre 2 actions est supérieur à `max-inactivity` et la durée de session résultante est supérieure à `max-session-length` auquel cas on coupe la session en deux.

🔗 * Utilisation de `Scenario.json` pour préparer le nettoyage des traces

`scenario.py`

Ce script regarde les doublons des ID de sons à partir du fichier `Scenario.json` afin de le mettre à jour, car il constitue un fichier source pour la suite des traitements

Données d'entrée

Il prend en entrée le fichier `Scenario.json` qui se trouve dans les `Traces brutes`

Données de sortie

Il renvoie, en sortie, un fichier `doublon_sonId.json` qui se situe dans le dossier des `Traces brutes`

`verif_sonid.py`

Le script `verif_sonid.py` permet de vérifier les ID des sons dans les traces LUCIOLE à partir du fichier source `Scenario.json`

Données d'entrée

On prend en entrée des fichiers JSON qui se situent dans les dossiers `Fluence_id_` du dossier `Traces transformées` et on les compare aux ID des sons qui se trouvent dans le `Scenario.json` du dossier `Traces brutes`

Données de sortie

En sortie, il renvoie le fichier `sons_manquants.json` dans le dossier `Traces transformées`. Il permet de voir tous les sons manquants ou ceux qui posent problème, d'après le `Scenario.json`

`luciole_specs.py`

Pour un exemple d'usage [S214](#)

nettoyage.py

Ce script permet de procéder au nettoyage des traces afin de résoudre les problèmes repérés dans celles-ci. Il utilise les classes et les méthodes créés dans l'utilitaire [luciole_specs.py](#)

Données d'entrée

Le script prend en entrée les traces transformées en JSON dans le dossier [Traces transformées](#)

Données de sortie

Le script `nettoyage.py` est appelé à partir de sa fonction `cleanup_json`, qui rassemble tous les traitements effectués, dans [fluence_id.py](#) afin de procéder à la désambiguïsation des traces et générer les fichiers JSON dans le dossier [Traces transformées](#)

session_tools.py

Le script gère toutes les fonctions de gestion des sessions :

- fusionne les doublons de sessions ;
- calcule la longueur d'une session ;
- calcule le temps d'inactivité d'une session.

Il est appelé à partir du script [fluence_id.py](#) et prend les mêmes données d'entrée et de sortie



2. JSON → CSV

indic_glob.py

Le script `indic_glob.py` sert à calculer les indicateurs qui se trouvent dans le ticket #3. Il utilise la classe `CsvBldr` de l'utilitaire [csv_builder.py](#)

Données d'entrée

Le script prend en entrée les traces transformées en JSON dans le dossier [Traces transformées](#)

Données de sortie

En sortie, il génère des fichiers CSV `indicateurs_globaux_[annee].csv` dans le dossier [Traces transformées](#)

autoeval.py

Ce script vérifie les sons des activités des traces des autoévaluations à partir du fichier `autoeval.json` situé dans le dossier [Traces brutes](#) et calcule les ID de sons des activités d'autoévaluations en se servant de la classe `CsvBldr` de l'utilitaire [csv_builder.py](#)

Données d'entrée

Il prend en entrée les traces transformées en JSON dans le dossier [Traces transformées](#)

Données de sortie

En sortie, il génère des fichiers CSV `autoevals_[annee].csv` dans le dossier [Traces transformées](#)

Annexe 2

Documentation pour les activités abandonnées

###ActiviteAbandon = false et est mal étiquetée :

1/ activités sans actions

2/ activités avec peu d'actions et le même typeSon

exemple : typeSon="consigne" + activiteScore= 0 (hormis les activités de narration qui n'ont aucune étoile) pour les 3 actions --> activiteAbandon devrait être = true

Lorsque activite Abandon=false et est correctement détectée :

1/ S'il y a une seule action dans l'activité et que le typeSon="intro" activiteType="Narration" (introduction de l'activité)

exemple :

```
{
  "actions": [
    {
      "sonId": "3731",
      "actionId": 1,
      "actionType": "stimulus",
      "actionTimestamp": 1517564720,
      "actionDeclencheur": "Auto",
      "typeSon": "intro"
    }
  ],
  "idMission": 0,
  "activiteId": 3724,
  "activiteEnd": 1517564744,
  "activiteType": "Narration",
  "activiteBegin": 1517564720,
  "activiteScore": 0,
  "activiteAbandon": false,
  "activiteOrigine": "Scenario"
```

2/ S'il y a une seule action dans l'activité et que le typeSon="conclu" et activiteType="Narration" (conclusion de l'activité)

exemple :

```
{
  "actions": [
    {
      "sonId": "3689",
      "actionId": 1,
      "actionType": "stimulus",
      "actionTimestamp": 1517564705,
```

```
        "actionDeclencheur": "Auto",
        "typeSon": "conclu"
    }
],
    "idMission": 0,
    "activiteId": 3684,
    "activiteEnd": 1517564720,
    "activiteType": "Narration",
    "activiteBegin": 1517564705,
    "activiteScore": 0,
    "activiteAbandon": false,
    "activiteOrigine": "Scenario"
},
```

3/ Lorsqu'on retrouve plusieurs actions avec différents typeSon, on s'appuie sur les activiteBegin et activiteEnd pour calculer le temps par activité (comme l'exemple ci-dessous : 65 secondes)

Table des matières

Remerciements	3
Sommaire.....	5
Introduction	7
PARTIE 1 - PRESENTATION DU STAGE ET CONTEXTE D'ACCUEIL	9
CHAPITRE 1. CONTEXTE D'ACCUEIL	10
1. Le Laboratoire d'InfoRmatique en Image et Systèmes d'information (LIRIS)	10
2. Le laboratoire Interactions, Corpus, Apprentissages, Représentations (ICAR)	11
CHAPITRE 2. PRESENTATION DU PROJET	12
1. Le projet Fluence e-FRAN	12
2. Les applications du projet Fluence	12
2.1. EVASION	12
2.2. ELARGIR	13
2.3. LUCIOLE	13
3. Le protocole expérimental du projet Fluence.....	14
CHAPITRE 3. OBJECTIF DU STAGE	16
1. Le stage : le projet LUCIOLE	16
2. Missions et tâches	16
PARTIE 2 - LE JEU LUCIOLE CONCEPTION, FONCTIONNALITES ET DONNEES	
COLLECTEES.....	18
CHAPITRE 4. LES FONCTIONNALITES DU JEU LUCIOLE.....	20
1. Se connecter à LUCIOLE	20
2. Structure du jeu LUCIOLE	20
2.1. Les missions.....	20
2.2. La structure des missions	21
2.3. Les activités	22
2.4. Les modalités de progression dans le jeu.....	23
2.5. Le déroulement des sessions de jeu	24
CHAPITRE 5. LES DONNEES COLLECTEES	26
1. Les mediaWiki	26
2. Tests et expérimentations	27
3. Définition des concepts utilisés.....	28
3.1. Qu'est-ce qu'une trace dans les EIAH ?	28
3.2. Le principe d'un Système à Base de Traces modélisées (SBTm).....	29
3.3. Qu'est-ce qu'une trace d'interaction dans LUCIOLE ?.....	30
3.4. Qu'est-ce qu'un indicateur ?	31
PARTIE 3 - ANALYSE DES TRACES LUCIOLE	33
CHAPITRE 7. LA PRISE EN MAIN DES TRACES.....	34

1. Modélisation des traces LUCIOLE	34
2. Exportation des traces LUCIOLE	36
2.1. Les traces brutes.....	37
2.2. Découpage des fichiers XML	38
3. Formalisme des traces en JSON.....	38
CHAPITRE 8. TRANSFORMATION DES TRACES LUCIOLE	40
1. Exportation des traces XML en JSON	40
2. Extraction des traces EVASION et ELARGIR.....	40
CHAPITRE 9. DESAMBIGUÏSATION DES TRACES LUCIOLE.....	41
1. Fusion des indicateurs	41
2. Réécriture des traces	42
2.1. Préparation des traces à l'aide d'un fichier source	42
2.2. Traitement des actions dans les traces	43
3. Gestion des sessions de jeu	44
CHAPITRE 10. CALCUL DES INDICATEURS	46
1. Le constructeur CSV	46
2. Création des routines pour le calcul des indicateurs	46
2.1. Les traces des autoévaluations	47
2.2. Les indicateurs globaux	47
2.3. L'usage du carnet de notes.....	48
3. Récapitulatif du travail réalisé et les difficultés rencontrées.....	49
3.1. Les contraintes rencontrées.....	49
4. Perspectives de l'analyse des traces LUCIOLE et extensions possibles.....	50
Conclusion et bilan personnel.....	53
Bibliographie	55
Sitographie.....	56
Sigles et abréviations utilisés.....	57
Table des illustrations.....	58
Table des annexes.....	59
Table des matières	66

MOTS-CLÉS : Jeu sérieux, dimension ludique, apprentissage, traces d'interaction, LUCIOLE.

RÉSUMÉ

Les jeux sérieux sont de plus en plus utilisés pour l'acquisition d'une langue étrangère et ce, dès le cycle 2. Toutefois, de nombreuses questions restent en suspens en ce qui concerne les protocoles expérimentaux et l'intérêt de ces jeux pour l'apprentissage. C'est ainsi que des chercheurs et pédagogues se sont réunis pour concevoir un jeu sérieux, LUCIOLE. Le jeu LUCIOLE a pour vocation de développer la compréhension orale de l'anglais chez les élèves de cycle 2 et a été testé pendant trois années. S'appuyant sur une dimension ludique, les premiers résultats des post-tests ont montré un effet de LUCIOLE dans les apprentissages qu'il convient d'affiner en les croisant avec les traces d'interaction. Dans ce travail, nous avons tenté d'analyser les traces LUCIOLE en définissant des routines pour le calcul des indicateurs. Pour cela, nous avons désambiguïsé et restructuré les traces en vue de mesurer les effets de LUCIOLE en lien avec les pré et post-test pour la validation du jeu.

KEYWORDS : Serious game, playful dimension, learning, interaction traces, LUCIOLE.

ABSTRACT

Serious games are increasingly being used for foreign language acquisition, starting in Cycle 2. However, many questions remain unanswered regarding experimental protocols and the value of these games for learning. This is why researchers and educators have come together to design a serious game, LUCIOLE. The LUCIOLE game aims to develop the oral comprehension of english in cycle 2 pupils and has been tested for three years. Based on a playful dimension, the first results of the post-tests showed an effect of LUCIOLE on learning that needs to be refined by cross-referencing them with the interaction traces. In this work, we tried to analyse the LUCIOLE traces by defining routines for the calculation of indicators. To do this, we disambiguated and restructured the traces in order to measure the effects of LUCIOLE in relation to the pre- and post-test for the validation of the game.