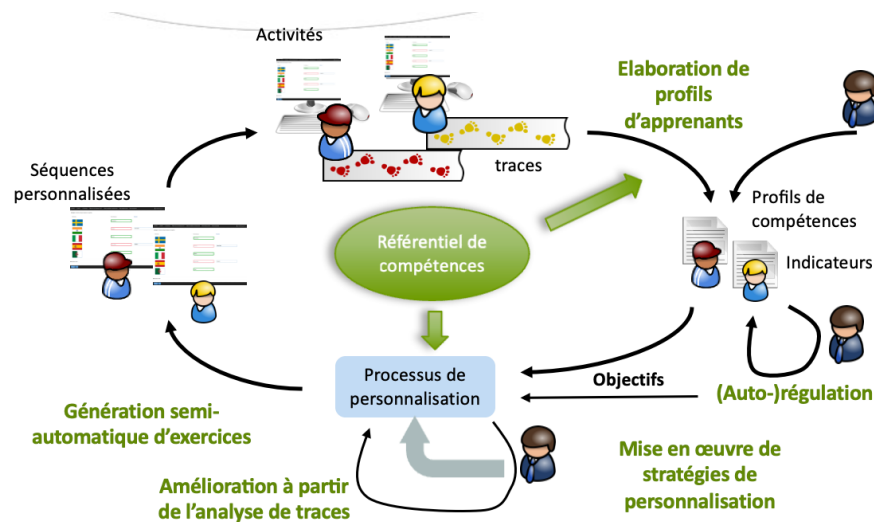


M2 IA – UE EIAH

TP ADAPTIVE LEARNING

L'objectif de ce TP est de vous confronter aux différentes étapes du cycle de personnalisation vu en cours et redonné ci-dessous.



Pour cela, nous allons exploiter le jeu sérieux SPY, proposé par le Mathieu Muratet du laboratoire LIP6, et les jeux de traces mis à disposition de la communauté EIAH.

LA PENSEE INFORMATIQUE

Depuis quelques années, l'informatique disciplinaire réapparaît dans les programmes scolaires. On ne parle pas d'enseigner la programmation ou l'informatique, mais on parle de **pensée informatique**. Selon Wing (Wing, 2006), la pensée informatique met en jeu un répertoire de cinq capacités cognitives : (1) la pensée algorithmique, (2) l'abstraction, (3) l'évaluation, (4) la décomposition, et (5) la généralisation.

L'appropriation de ces compétences par les enseignants de l'école primaire est complexe à résoudre. De nombreux travaux tentent d'apporter des solutions en proposant des outils permettant aux enseignants de faire travailler leurs élèves, mais également de prendre en main la création de ressources pour travailler cette pensée informatique.

LE JEU SPY

SPY est un jeu sérieux d'apprentissage sur le thème de la pensée informatique. Il a été conçu pour un public d'élèves de cycle 3 (CM1, CM2, 6ème). Le principe du jeu est de programmer un ou

plusieurs agents à l'aide d'actions simples afin de les déplacer sur une grille vers des positions particulières. Ces actions sont représentées sous forme de blocs que le joueur doit agencer en séquences.

Pour programmer le robot, le joueur bénéficie de blocs d'action (« Avancer », « Pivoter à gauche », « Pivoter à droite », « Attendre », « Activer un terminal » et « Faire demi-tour »), de blocs de contrôle (« Si Alors », « Si Alors Sinon », « Répéter n fois », « Tant que » et « Répéter indéfiniment »), de capteurs (« Mur en face », « Mur à gauche », « Mur à droite », « Passage en face », « Passage à gauche », « Passage à droite », « Ennemi en face », « Zone surveillée en face », « Porte détectée en face », « Terminal détecté sur ma position » et « Sortie détectée sur ma position ») et d'opérateurs (« Et », « Ou » et « Non »).

SPY contient plusieurs scénarios de jeu clés en main qui permettent aux enseignants de faire travailler leurs élèves sur des compétences de la pensée informatique : observer et modéliser la simulation (abstraction) ; décomposer sa stratégie en sous-étapes (décomposition) ; déterminer la meilleure solution (évaluation) ; planifier les actions à réaliser (pensée algorithmique) ; concevoir des solutions permettant de résoudre plusieurs problèmes (généralisation).

Outre ces scénarios, SPY contient un éditeur de niveau qui permet aux enseignants de créer leurs propres niveaux adaptés à leurs élèves et de les agencer à l'aide de l'éditeur de scénario.

Les scénarios inclus actuellement dans SPY sont nommés : Explorateur, Collaborateur, Répétiteur, Sélectionneur, Infiltration, BlocklyMaze. Le détail de ces scénarios est présenté dans [l'article \(Muratet, 2023\)](#).

PRISE EN MAIN

Le jeu est disponible à cette adresse : https://spy.lip6.fr/index_fr.html

Sans vous amuser ;-), nous allons prendre en main le jeu SPY pour comprendre les objets manipulés et pouvoir analyser les traces des étudiants.

Lancer le jeu, faites le tutoriel en choisissant comme référentiel de compétences "SPY".

Faites ensuite un autre scénario, par exemple "Sélectionneur" avec le même référentiel de compétences.

LES REFERENTIELS DE COMPETENCES

Vous avez pu voir qu'à chaque jeu est associé un ensemble de compétences. Actuellement trois référentiels sont intégrés au jeu : le PIAF (Parmentier et al., 2020), le [CRCN](#) et celui de SPY.

[Seymour Papert \(1971\)](#) a posé les bases d'une pensée informatique. Depuis une quinzaine d'années, la prise de conscience collective de l'importance de définir un cadre de formation à ce qui s'apparente à la pensée informatique a amené à la proposition de différents référentiels de compétences sur ce sujet.

Le PIAF (Pensée Informatique et Algorithmique dans l'enseignement Fondamental) est un référentiel issu du projet PIAF soutenu par l'Union Européenne et visant à développer l'apprentissage de la pensée informatique et algorithmique dans l'enseignement fondamental. Ce projet rassemble des chercheurs en sciences de l'éducation et en informatique, provenant de quatre pays (Allemagne, Belgique, France et Luxembourg), autour du thème de la formation des enseignants.

Le cadre de référence des compétences numériques ([CRCN](#)) est inspiré du cadre européen DigComp 2.1 ([Digital Competence Framework for Citizens, 2017](#)). Il est entré en vigueur à la rentrée scolaire 2019 et est valable de l'école primaire à l'université. Un bilan de la maîtrise des compétences numériques des élèves est réalisé en classe de cours moyen deuxième année (CM2) et en classe de sixième pour le cycle 3. Une certification du niveau de maîtrise des compétences numériques est délivrée à tous les élèves en fin de cycle 4 des collèges et en fin de cycle terminal des lycées. Le CRCN fixe 16 compétences numériques attendues dans 5 domaines d'activité et huit niveaux de maîtrise de ces compétences (article D. 121-1 du code de l'éducation).

LES TRACES D'APPRENTISSAGE

SPY partage les données d'interactions générées par le jeu. L'ensemble des interactions sont tracées au format [xAPI](#) et stockées sur un LRS (Learning Record Store). Les données sont accessibles *via* un LearningLocker.

Chaque trace au format xAPI est appelé *statement* et se compose d'un triplet *Acteur/Verbe/Objet* (Qui ? Fait quoi ? Sur quoi ?). Chaque nouvelle connexion au jeu génère un identifiant de session anonyme unique que le joueur peut noter en vue de recharger sa progression ultérieurement. Cet identifiant de session définit l'acteur du *statement*. Les verbes et les objets permettent de caractériser le contenu des interactions.

Les interactions actuellement tracées sont les suivantes : lancement du niveau, ouverture des dialogues, passage au dialogue suivant, passage au dialogue précédent, fermeture des dialogues, création d'une zone de programme, nettoyage d'une zone de programme, suppression d'une zone de programme, renommage d'une zone de programme, insertion d'un bloc, suppression d'un bloc, modification d'un bloc, exécution d'un programme, mise en pause de l'exécution, exécution pas à pas, arrêt de l'exécution d'un programme, fin du niveau, fermeture du niveau.

Certains *statements* contiennent des extensions qui permettent de stocker des informations complémentaires comme les conditions de fin du niveau (succès/échec, score) ou le contexte dans lequel un bloc a été ajouté/supprimé. Actuellement seule la partie de SPY manipulée par l'élève est tracée.

Toutes les informations permettant d'accéder aux traces du jeu sont disponibles à l'adresse suivante : <https://spy.lip6.fr/openTraces.html>. Il y a la base de traces, et le vocabulaire utilisé pour collecter les traces.

TRAVAIL A FAIRE

L'outil LearningLocker permet de générer des visualisations, de calculer des indicateurs et de télécharger les données.

Vous avez accès toutes les données, dont celles que vous venez de produire en jouant à SPY.

Prenez une des compétences associées au niveau que vous avez fait et essayer de proposer une solution pour identifier si un apprenant ayant travaillé avec SPY possède ou non cette compétence.

Testez sur plusieurs apprenants puis testez sur vos propres traces d'activité.

Écrivez un petit document décrivant la compétence choisie, la méthode retenue pour en calculer la maîtrise et les tests effectués pour valider votre approche. Indiquez éventuellement les données qu'il vous a manqué pour parfaire le résultat.

Complétez le document en indiquant comment vous pourriez exploiter vos résultats pour mettre en place de l'Adaptive Learning dans le jeu SPY.

Déposer sur Tomuss le rapport, l'export des traces sur lesquelles vous avez travaillé et votre code dans une archive.

Si vous avez bien travaillé, vous aurez un bonus sur la note finale de l'UE...

RÉFÉRENCES

Muratet, M. (2023). SPY : Un jeu sérieux partagé pour étudier l'apprentissage de la pensée informatique. Actes de l'atelier "Apprendre la Pensée Informatique de la Maternelle à l'Université" Brest, France.

Papert, S. (1971). Teaching Children Thinking. Artificial Intelligence, Memo No. 247, Massachusetts Institute of Technology, A.I. Laboratory.

Parmentier, Y., Reuter, R., Higuette, S., Kataja, L., Kreis, Y., Duflot-Kremer, M., Laduron, C., Meyers, C., Busana, G., Weinberger, A. and Denis, B. (2020). PIAF: developing computational and algorithmic thinking in fundamental education. In: EdMedia+ Innovate Learning, Association for the Advancement of Computing in Education (AACE), pages 315–322.

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>