

Git4School : un tableau de bord pour assister la prise de décisions de l'enseignant lors des cours de génie logiciel

Jean-Baptiste Raclet, Franck Silvestre, Mika Pons

▶ **To cite this version:**

Jean-Baptiste Raclet, Franck Silvestre, Mika Pons. Git4School : un tableau de bord pour assister la prise de décisions de l'enseignant lors des cours de génie logiciel. STICEF (Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation), 2021, Numéro Spécial : Technologies pour l'apprentissage de l'Informatique de la maternelle à l'université, 28 (3), pp.1-20. 10.23709/sticef.28.3.2 . hal-03617760

HAL Id: hal-03617760
https://hal.science/hal-03617760

Submitted on 28 Mar 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.



Git4School : un tableau de bord pour assister la prise de décisions de l'enseignant lors des cours de génie logiciel

▶ **Jean-Baptiste RACLET** (IRIT, Université Toulouse 3), **Franck SILVESTRE** (IRIT, Université Toulouse 1), **Mika PONS** (IRIT, Université Toulouse 3),

■ **RÉSUMÉ** • Cet article présente Git4School, un tableau de bord pour les enseignants offrant des visualisations s'appuyant sur des données extraites des dépôts Git des apprenants, combinées à des informations contextuelles temporelles. Sur la base de plusieurs expérimentations, nous montrons un bon sentiment général des apprenants sur l'utilisation de Git dans un contexte éducatif. De plus, nous montrons comment Git4School peut être utile pour identifier les faiblesses des conceptions des situations d'apprentissage et cibler leur amélioration.

■ **MOTS-CLÉS** • enseignement du génie logiciel, conception de situation d'apprentissage, intervention pédagogique, prise de décision, tableau de bord, Git.

■ ***ABSTRACT*** • *This article presents Git4School, a dashboard for teachers providing visualizations based on data extracted from learners' Git repositories, combined with temporal contextual information. Based on several experiments, we show a good general feeling of learners about using Git in an educational context. We show how Git4School can be useful for identifying weaknesses in learning designs and for targeting their needs for improvement.*

■ ***KEYWORDS*** • *software engineering education, learning design, pedagogical intervention, decision-making, dashboard, Git.*

1. Introduction

Une conception de situation d'apprentissage[1] décrit une séquence d'apprentissage à l'échelle d'une partie ou d'une unité d'enseignement entière. Elle identifie les principaux acteurs impliqués (enseignants et apprenants), les tâches d'enseignement et d'apprentissage à accomplir ainsi que les ressources pédagogiques utilisées pour soutenir les activités de la séquence (Lockyer *et al.*, 2013). Ce concept est apparu au début des années 2000 dans le but de favoriser la mutualisation de scénarios pédagogiques exploitant les technologies numériques (Emin *et al.*, 2011).

Si les conceptions de situation d'apprentissage permettent de décrire une intention pédagogique, elles n'identifient cependant pas comment les apprenants participent réellement à cette conception, que ce soit pendant son développement ou *a posteriori*. Pour obtenir de telles informations, il est nécessaire de s'appuyer sur la collecte et l'analyse de traces d'utilisation (Choquet *et al.*, 2007), ce que l'on désigne plus récemment sous l'appellation *learning analytics*. Les *learning analytics* constituent un concept clé pour obtenir une perspective globale de l'impact des activités d'apprentissage (Wise, 2014). En effet, les *learning analytics* s'appuient sur le recueil d'ensembles de données correspondant aux traces du travail des apprenants afin de mesurer leur engagement et, éventuellement, d'extraire des preuves d'apprentissage. Dans le contexte de conception de situation d'apprentissage, la visualisation des données d'apprentissage peut soutenir la prise de décision de l'enseignant au cours d'une séquence : synthétisée sous la forme d'un tableau de bord (Schwendimann *et al.*, 2017), elle permet aux enseignants d'obtenir des informations en temps réel sur l'avancement des élèves dans leurs activités et permet ainsi d'adapter l'intervention pour rendre leur situation d'apprentissage plus efficace.

Cet article s'inscrit dans cette vision et l'aborde dans le contexte particulier de l'enseignement du génie logiciel. L'enseignement dans cette discipline de l'informatique combine des aspects théoriques abstraits avec des considérations très pratiques souvent outillées. En particulier, les travaux pratiques impliquent généralement l'utilisation d'un environnement de développement intégré avec un éditeur de code source, un compilateur et un débogueur. En plus de ces outils, des outils

[1] En se basant sur le TEL thesaurus, le terme anglophone « *learning design* » est traduit dans cet article par « conception de situation d'apprentissage ».

de gestion de versions de code source comme Git (https://git-scm.com/) qui sont largement utilisés dans l'industrie du logiciel (Bourque et Fairley, 2014), sont parfois utilisés pour collecter les travaux des étudiants. Dans cet article, nous explorons une manière originale de construire des visualisations s'appuyant sur des données extraites de dépôts Git combinées à des informations contextuelles temporelles complémentaires (par exemple, le travail a été effectué à l'intérieur ou à l'extérieur de la classe, avant ou après une intervention ou une activité particulière). Cela nous a amené aux deux questions de recherche suivantes :

– (**RQ1**) Les étudiants sont-ils capables, quel que soit leur niveau d'études, d'utiliser des outils de gestion de versions comme Git à des fins éducatives ?

– (**RQ2**) Dans quelle mesure un tableau de bord construit à partir de données extraites de Git combinées à des données contextuelles temporelles complémentaires peut-il aider l'enseignant à prendre des décisions pendant un cours ?

Pour répondre à (**RQ1**), nous avons mené une étude quantitative en contexte écologique sur les activités des étudiants relatives à l'utilisation de Git dans 5 cours répartis dans 4 niveaux d'études différents. Nous avons abordé (**RQ2**) d'abord en concevant le tableau de bord appelé Git4School et ensuite par une étude qualitative sur l'utilisation de Git4school en contexte écologique.

La section 2 présente un état de l'art des travaux relatifs aux tableaux de bord, à la prise de décision dans le contexte de la conception de situation d'apprentissage ainsi qu'à l'exploitation de données d'apprentissage extraites des outils de gestion de versions. La section 3 présente l'étude quantitative relative à l'utilisation de Git par les étudiants. La section 4 est consacrée à la description du tableau de bord Git4School tandis que la section 5 présente deux études de cas sur l'utilisation de Git4School dans les cours en face à face. L'article se termine par une synthèse des réponses apportées à (**RQ1**) et (**RQ2**) et propose des pistes d'amélioration.

2. Travaux connexes

Faisant partie de la grande famille des outils d'analyse de données, les outils et solutions issus des *learning analytics* sont efficaces lorsqu'ils aident les personnes à prendre des décisions et à agir (Van Harmelen et

Workman, 2012). Pour tenir cette promesse d'efficacité, nous suivons l'approche préconisée par Gašević *et al.* (Gašević *et al.*, 2015) qui défendent que les *learning analytics* devraient être fortement ancrées dans les cadres pédagogiques issus de la recherche en éducation.

Comme défini par Lockyer *et al.* (Lockyer *et al.*, 2013) une conception de situation d'apprentissage décrit la séquence des tâches, des ressources et des supports d'apprentissage qu'un enseignant construit pour les cours des étudiants. Dans leurs travaux, les auteurs indiquent, de plus, que les objectifs d'apprentissage et le plan pédagogique peuvent être évalués au moyen de *learning analytics*. Celles-ci fournissent les informations manquantes sur la manière dont les étudiants participent au plan initial et peuvent ensuite aider les enseignants et les formations à évaluer l'impact des activités d'apprentissage.

Plus axée sur la prise de décision des enseignants pendant les séquences de cours, la communauté de recherche sur l'apprentissage collaboratif supporté par l'informatique a développé des idées similaires autour du concept de technologie d'orchestration (Tissenbaum et Slotta, 2019), (Tchounikine, 2013, p 501) : « *des technologies qui fournissent aux enseignants des moyens de surveillance ou d'intervention* ». Les technologies d'orchestration prennent souvent la forme de tableaux de bord aidant les enseignants à obtenir un retour d'information pour savoir quand et où intervenir (Tissenbaum et Slotta, 2019).

Enfin, Wise (Wise, 2014) a proposé le concept de conception d'intervention pédagogique fondée sur les *learning analytics*[2] en tant qu'« *efforts systématiques pour intégrer l'utilisation des learning analytics comme partie productive des pratiques d'enseignement et d'apprentissage dans un contexte éducatif donné* ». Notre proposition s'inscrit directement dans la lignée des travaux de Wise puisque nous utilisons les *learning analytics* et plus précisément les visualisations afin d'aider les enseignants à réaliser des interventions pendant leurs cours en face à face. Le tableau de bord que nous avons conçu et que nous décrivons plus loin s'inspire principalement des recherches menées par Bakharia *et al.* (Bakharia *et al.*, 2016, p. 330) qui visent à « créer des représentations plus significatives des données pour les enseignants ». Dans notre cas, nous avons combiné deux dimensions du cadre défini par Bakharia *et al.* : la dimension temporelle et

[2] En anglais, « *pedagogical learning analytics intervention design* ».

la dimension dynamique de cohorte, les données provenant d'une part de l'outil de gestion de version du code source et, d'autre part, des données fournies par l'enseignant.

Il existe dans la littérature plusieurs travaux concernant l'extraction de métriques à partir des journaux d'un outil de gestion de version du code. Contrairement à notre approche qui analyse les messages de *commits* porteurs d'informations sémantiques (étape de conception de l'apprentissage au moment de la réalisation d'un exercice identifié), ces travaux s'intéressent plutôt aux informations sur la volumétrie des *commits*, leur fréquence, leur temporalité par rapport à une échéance.

La plupart des travaux existants considèrent les outils de gestion de version centralisés appelés CVS (Liu *et al.*, 2004), (Mierle *et al.*, 2005) ou bien alors SVN (Glassy, 2006), (Jones, 2010), (Kay *et al.*, 2006), (Poncin *et al.*, 2011). À notre connaissance, la seule approche qui s'appuie sur Git est (Robles et Gonzalez-Barahona, 2013). Les objectifs de ces travaux sont variés : le suivi des étudiants (Glassy, 2006), l'analyse de la collaboration entre les différents membres d'un groupe de travail (Jones, 2010), (Kay *et al.*, 2006), (Mittal et Sureka, 2014), (Robles et Gonzalez-Barahona, 2013) avec éventuellement la corrélation avec les résultats obtenus (Liu *et al.*, 2004) ou une prédiction des performances (Mierle *et al.*, 2005). Aucun de ces travaux ne tente de combiner les données extraites des dépôts de code avec d'autres données contextuelles pour la prise de décision dans les interventions pédagogiques comme nous le présentons dans cet article.

3. Évaluation quantitative

L'objectif principal de l'étude quantitative qui est présentée dans cette section est de fournir des éléments de réponse à la question de recherche (**RQ1**).

3.1. Contexte des différentes expérimentations

Durant l'année universitaire 2019-2020, les concepteurs de Git4School ont demandé à leurs étudiants d'utiliser Git ainsi que le service d'hébergement GitHub (https://github.com), dans le cadre de 5 enseignements différents concernant le développement logiciel. Plus précisément, les étudiants devaient soumettre leurs travaux en effectuant des opérations de *commit* au sein d'un dépôt Git individuel. Ceci a entrainé la création de 180 dépôts Git par 156 étudiants, contenant 3 082 *commits*.

Le tableau 1 indique le nombre de dépôts créés par les étudiants en fonction du niveau du diplôme (selon la classification *International standard classification of education* (ISCED, 2012) qu'ils préparent et leur année d'études.

Tableau 1 · Nombre de dépôt en fonction du niveau d'études

Niveau ISCED	5 (DUT, licence pro)		7 (master)	
Année d'étude	2ème	3ème	1ère	2ème
Nombre de dépôts	73	16	55	36
Nombre d'étudiants	49	16	55	36

Parmi tous les *commits* effectués par les étudiants, on peut distinguer des *fix commits* qui comportent un message imposé par l'enseignant dans son énoncé afin d'indiquer la fin d'un exercice donné.

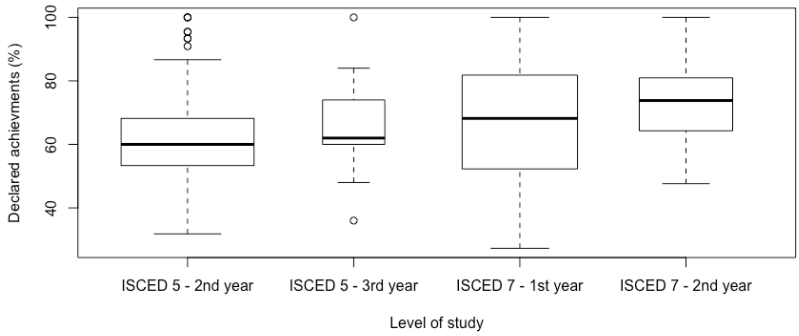


Figure 1 · Part de « *fix commits* » sur l'ensemble des « *commits* » en fonction de l'année d'étude

Dans les dépôts mentionnés ci-dessus, 2424 *fix commits* ont été identifiés. La figure 1 montre leur distribution selon le niveau du diplôme et l'année d'étude de l'étudiant.

Précisons que, même lorsque les étudiants n'ont pas effectué tous les exercices, tous ont réussi, quel que soit leur niveau d'études, à utiliser Git et GitHub en suivant la procédure imposée pour soumettre leurs travaux.

3.2. Analyse de l'enquête auprès des étudiants

À la fin de chaque cours, nous avons mené une enquête auprès des étudiants pour évaluer leur ressenti envers Git et GitHub. Le tableau 2 énumère différentes affirmations que les étudiants ont été invités à

évaluer en leur donnant une note selon une échelle de Likert à 7 niveaux : « 1 » signifiant « Pas du tout d'accord » et « 7 » signifiant « Tout à fait d'accord ».

Tableau 2 · Affirmations évaluées par les étudiants

#	Affirmations évaluées
1	J'ai trouvé qu'il était facile d'utiliser Git et GitHub pour soumettre mon travail.
2	J'ai trouvé l'utilisation de Git et GitHub adaptée à la soumission de mon travail.
3	J'aimerais que mes travaux futurs soient également soumis en utilisant Git et GitHub.
4	J'ai trouvé que l'utilisation de Git et GitHub ajoutait de la complexité au travail.
5	J'ai trouvé intéressante l'utilisation de Git et GitHub lors du travail.
6	J'ai trouvé motivante l'utilisation de Git et GitHub lors du travail.

La figure 2 représente la répartition des 61 réponses (39% des étudiants participant aux expérimentations) que nous avons obtenues. Les affirmations 1 et 4 ont été demandés dans le but d'évaluer la difficulté perçue relative à l'utilisation de Git et GitHub. Avec une médiane respectivement de 6 et 2 et un IQR resserré autour des médianes, l'enquête révèle que l'utilisation de Git et GitHub est largement considérée comme facile et n'augmente pas la complexité du travail à effectuer.

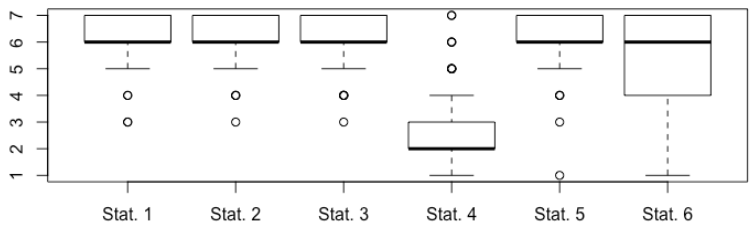


Figure 2 · Distribution des réponses des étudiants

Ensuite, les affirmations 2 et 3 permettent d'évaluer l'utilité perçue de Git et GitHub pour la soumission des travaux. Là encore, avec des médianes de 6 et un IQR resserré autour des médianes, l'enquête indique que les élèves considèrent Git et GitHub comme étant adaptés pour soumettre leurs travaux et sont prêts à utiliser ces outils pour de futurs travaux.

Enfin, les affirmations 5 et 6 visent à évaluer dans quelle mesure les étudiants considèrent Git et GitHub comme des outils intéressants et motivants. Les deux médianes se situent à 6. L'IQR pour la déclaration 6 indique une plus grande distribution des réponses par rapport à la motivation. On peut considérer que les étudiants trouvent très majoritairement intéressant l'utilisation de Git et GitHub alors qu'une grande majorité la trouve motivante. La mobilisation d'outils utilisés dans le contexte professionnel s'intègre pleinement dans un des six leviers identifiés par Poumay (Poumay, 2014) pour donner du sens aux apprentissages et augmenter l'intérêt des étudiants. Les résultats obtenus sur les affirmations 5 et 6 confortent cette proposition.

En outre, il est possible de vérifier que les réponses données par les étudiants ne dépendent pas de leur niveau d'études. À cette fin, pour chaque affirmation A_i de l'enquête, nous effectuons le test exact de Fisher pour tester la dépendance entre la variable du niveau ISCED et la variable R_i^2 définie comme suit :

$Soit\ R_i^7\ l'évaluation\ donnée\ par\ un\ étudiant\ à\ l'affirmation\ A_i.$

$$Pour\ i\ \in \{1,2,3,5,6\}:$$

$$R_i^2 = \begin{cases} 1\ si\ R_i^7 \leq 4 \\ 2\ si\ R_i^7 > 4 \end{cases} \qquad R_4^2 = \begin{cases} 1\ si\ R_4^7 \leq 3 \\ 2\ si\ R_4^7 > 3 \end{cases}$$

Le tableau 3 indique la *p-valeur* calculée avec le test exact de Fisher pour évaluer la dépendance entre la variable du niveau ISCED et la variable R_i^2 pour chaque énoncé.

Tableau 3 · Résultats du test exact de Fisher

Affirmation	1	2	3	4	5	6
p-value	0.44	1	1	0.35	0.37	1

Toutes les *p-valeurs* sont très supérieures à 0,05, indiquant que le niveau d'études est indépendant de la réponse aux différentes questions posées.

3.3. Synthèse de l'évaluation quantitative

L'évaluation quantitative présentée fournit des réponses à la question de recherche (**RQ1**) concernant l'obligation pour les étudiants d'utiliser Git et GitHub à des fins pédagogiques quel que soit leur niveau d'études. En effet, elle révèle, d'une part, l'utilisation concrète de Git et GitHub par

tous les étudiants inscrits dans les cinq cours concernés par les expérimentations et, d'autre part, un bon ressenti général des étudiants concernant la facilité, l'utilité, l'intérêt et la motivation à utiliser ces deux outils dans un contexte éducatif, quel que soit leur niveau d'études.

4. Le tableau de bord Git4School

Git4School (https://git4school.firebaseapp.com/) est un tableau de bord, destiné aux enseignants, permettant de visualiser les progrès des élèves.

Sur la page web dédiée à chaque dépôt, GitHub propose un onglet « Insights » qui fournit diverses mesures comme la temporalité, la volumétrie et la répartition des *commits* entre les différents contributeurs du dépôt. Dans cette section, nous verrons que Git4School permet de visualiser des informations davantage sémantiques sur le travail effectué par les étudiants.

4.1. Hypothèses sur le contexte d'utilisation

Les étudiants travaillent individuellement pour développer une partie logicielle dans laquelle les principales fonctionnalités peuvent être mises en œuvre avec des notions de programmation ciblées. Lorsque les étudiants ont produit leur réponse à un exercice, ils effectuent un *commit* avec un message spécifique indiquant que l'exercice a été achevé et un *push* dans un dépôt privé GitHub qui leur est attribué. L'attribution de tels dépôts est facilitée par l'initiative GitHub Classroom (https://classroom.github.com/) qui automatise la création de dépôts Git avec des droits d'accès configurables tout en incluant un code de démarrage. Dans sa conception de la situation d'apprentissage, l'enseignant peut avoir prévu des tâches à accomplir par l'étudiant comme par exemple le travail de développement en autonomie ou la confrontation des solutions entre étudiants via une revue du code par les pairs. Une solution fournie par l'enseignant peut également être publiée. Dans ce qui suit, ces événements seront appelés « jalons ».

4.2. Les visualisations proposées par Git4School

La plateforme Git4School est une application web dont le code source (https://github.com/git4school/git4school-visu) est publié sous la licence Apache 2.0. Son utilisation nécessite que l'enseignant dispose d'un compte GitHub. L'API REST de GitHub (https://developer.github.com/v3/) est

utilisée pour extraire des informations sur les *commit* effectués par les étudiants dans leurs dépôts.

Les deux seules hypothèses de travail qui conditionnent l'utilisation de Git4School sont d'une part, l'utilisation de GitHub par les étudiants et, d'autre part, le respect de la bonne pratique consistant à effectuer un *commit* après chaque achèvement d'un exercice. Git4School est donc très peu invasif à la fois d'un point de vue méthodologique mais aussi en termes d'outils : la plateforme peut être utilisée quel que soit l'environnement de travail (système d'exploitation, IDE) et le langage de programmation.

L'enseignant, une fois connecté à Git4School, fournit un ensemble d'éléments de configuration à travers l'interface dédiée ou via un fichier au format JSON contenant les informations suivantes :

– La liste des dépôts d'étudiants hébergés sur GitHub.

– Une liste d'identifiants pour les différents exercices qui seront recherchés dans les messages de *commit*s. La présence d'un identifiant est interprétée comme l'achèvement par l'étudiant de l'exercice correspondant.

– La date et la durée des séances de travail en face à face.

– Pour chaque exercice, des jalons peuvent être mentionnés : revue par les pairs, publication d'une solution, remédiation, etc.

Ensuite, l'enseignant peut accéder à trois visualisations. Sur chacune d'elles, les couleurs portent des informations sémantiques : le vert indique un exercice réalisé en autonomie avant toute intervention déclenchée par l'enseignant ; l'orange indique un exercice réalisé après un jalon de type « évaluation par les pairs » ; enfin, le rouge indique un exercice réalisé après la publication d'une solution par l'enseignant.

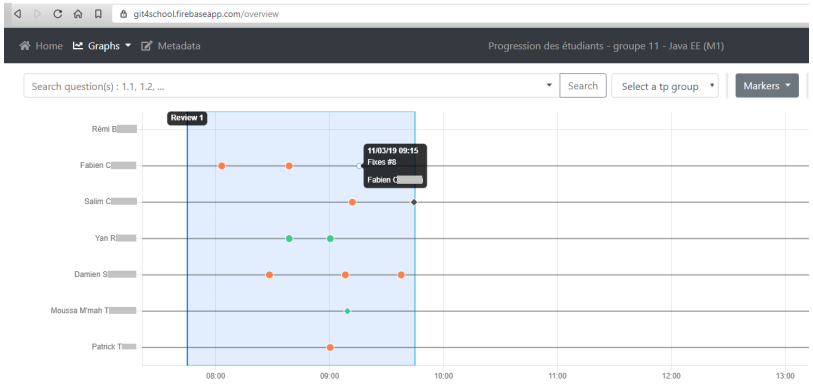


Figure 3 · Visualisation du progrès global dans Git4School

La première visualisation (Figure 3) permet de visualiser pour chaque étudiant, en ordonnée, tous ses *commits* dans le temps, en abscisse. Les temps correspondant aux séances en face à face (fond bleu clair) et aux jalons (lignes verticales) sont également identifiés. Les *commits* sont représentés avec des couleurs différentes selon qu'ils sont effectués avant ou après un jalon.

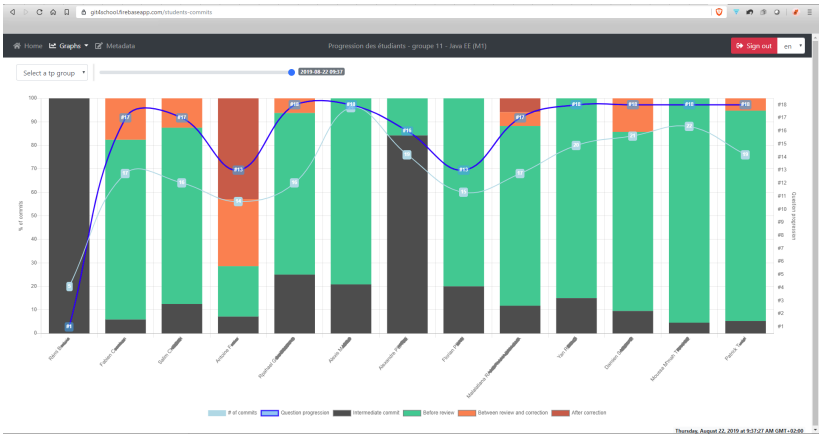


Figure 4 · Visualisation de la quantité de *commits* de chaque type pour chaque élève dans Git4School

La deuxième visualisation (Figure 4) indique, sous forme d'un diagramme en bâtons, la quantité de *commits* de chaque étudiant colorée en fonction de leur position temporelle par rapport aux différents types

de jalons. Pour chaque étudiant apparaît aussi l'identifiant du dernier exercice achevé et le nombre de *commits* exécutés.

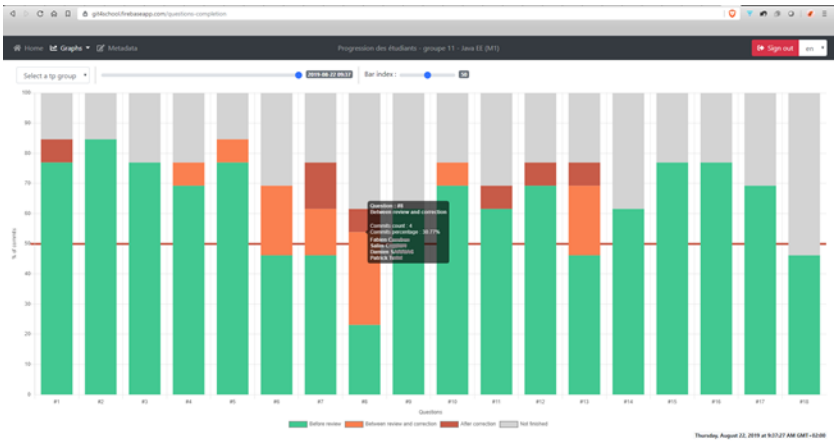


Figure 5 · Visualisation du niveau achèvement des exercices dans Git4School

La troisième visualisation (Figure 5) permet de visualiser sous forme d'un diagramme en bâtons, pour chaque exercice, le nombre d'étudiants qui l'ont réalisé et dans quelles circonstances (*commits* lors d'un travail en autonomie, après revue par les pairs et après publication de la solution).

5. Évaluation qualitative des expériences

Nous nous concentrons maintenant sur 2 des 5 expériences mentionnées dans l'évaluation quantitative. Ces études de cas ont eu lieu dans le cadre de cours d'informatique dispensés à différents niveaux de l'enseignement supérieur et suivant différents modèles d'apprentissage. Les deux études visaient à explorer dans quelle mesure les visualisations fournies par Git4School peuvent effectivement aider les enseignants à prendre des décisions au cours des différentes séquences composant leur cours. Nous fournissons ainsi des éléments de réponses en relation avec (**RQ2**).

5.1. Présentation des deux études de cas sélectionnées

5.1.1. Étude de cas 1

La première étude a eu lieu en deuxième année de Master en génie logiciel dans le cadre d'un cours consacré à l'API de persistance de Java (JPA). Un groupe de 36 étudiants était inscrit à ce cours. Ils ont suivi 10 interventions planifiées de 2 heures chacune, au cours desquelles ils ont mis en pratique les connaissances théoriques qu'ils avaient reçues précédemment sur JPA. Ces interventions ont suivi un protocole pédagogique original développé par l'équipe pédagogique au cours des dernières années pour couvrir d'une part l'acquisition de compétences individuelles en programmation et d'autre part l'acquisition de compétences sociales liées à la programmation. Ainsi, les interventions se sont déroulées comme une succession de cycles composés de 3 phases ordonnées comme suit :

– Phase de travail en autonomie : chaque élève travaille individuellement sur un même projet spécifié par un ensemble de tests unitaires et d'intégration. Les résultats et les messages fournis lors des tests automatisés donnent aux élèves un *feedback* qui facilite leur progression vers chaque objectif d'apprentissage.

– Phase de revue par les pairs : les étudiants participent à une séquence de revue par les pairs durant laquelle ils sont invités à présenter et à argumenter en faveur de leur solution à un exercice donné ou à présenter leur compréhension de l'utilisation de l'API. Cette phase de revue par les pairs est orchestrée à l'aide de la plateforme elaastic (https://elaastic.irit.fr/elaastic-questions) qui met en œuvre certains principes d'apprentissage actif présentés (Silvestre *et al.*, 2015).

– Phase de restitution : l'enseignant présente les résultats et commente une ou plusieurs solutions ou explications. Il anime les discussions.

À la fin de chaque contribution significative au projet, il a été demandé à chaque étudiant de réaliser un *commit* et un *push* de son travail sur son dépôt GitHub dès que tous les tests passaient avec succès sur la partie concernée du projet. L'indication de soumettre le travail faisait partie de l'énoncé du sujet.

5.1.2. Étude de cas 2

La deuxième étude de cas a eu lieu en deuxième année d'un cours de Diplôme Universitaire Technologique (DUT) informatique visant à

acquérir des connaissances et des compétences en matière de développement d'applications web avec le langage de programmation PHP (https://www.php.net). Deux groupes d'environ 25 étudiants ont participé au cours. La plupart d'entre eux étaient novices sur le sujet enseigné. Pendant 5 semaines, l'enseignant a utilisé Git4School. Durant cette période, chaque semaine, les étudiants ont reçu un cours théorique d'une heure sur l'utilisation d'une bibliothèque pour connecter des applications *PHP* avec des bases de données relationnelles ou sur la pertinence du modèle de conception modèle-vue-contrôleur dans une application web. Ensuite, ils ont mis en pratique ce contenu théorique pendant une session d'une heure trente suivant un modèle d'apprentissage assez simple : ils devaient essayer de compléter les exercices jusqu'à ce que le professeur décide d'interrompre le travail individuel pour donner des conseils ou présenter la solution d'un exercice ou de l'ensemble de l'exercice. Les étudiants étaient encouragés à s'entraider, mais aucun protocole d'enseignement rigoureux n'a été établi pour parvenir à cette instruction par les pairs, contrairement à la première étude de cas.

Pour permettre l'utilisation de Git4School, il a été demandé aux étudiants de réaliser un *commit* et *push* de leur travail à la fin de chaque exercice. Les indications de *commit* et *push* du travail ont été explicitement énoncées dans les sujets des différents travaux pratiques. Les exercices n'étant pas accompagnés de tests automatiques, les étudiants décidaient eux-mêmes si l'exercice était terminé. Ils pouvaient éventuellement demander à l'enseignant de le vérifier.

5.2. Résultats et discussions

Nous discutons maintenant des observations qui ont émergées de l'utilisation de Git4School au cours des études de cas décrites précédemment.

Déclenchement des interventions planifiées

La visualisation du taux d'achèvement par exercice a été très utile pour déclencher les interventions prévues dans les situations d'apprentissage (phases de revue par les pairs pendant les séquences relatives à l'étude de cas 1 et phases de restitution dans les deux études de cas). En effet, il est possible de déclencher une intervention à partir d'un seuil configurable de taux d'achèvement des élèves. Comme le montre la figure 5, le seuil personnalisé apparaît sur la visualisation sous la forme

d'une ligne horizontale rouge. Par exemple, pour un exercice donné, l'enseignant de la première étude de cas déclenchait la phase de revue par les pairs lorsque plus ou moins 60% des élèves avait terminé l'exercice.

Obtenir un feedback sur l'efficacité d'une intervention planifiée

Les trois visualisations proposées par Git4School affichent les *commits* des élèves avec une couleur fonction de la position dans le temps du *commit* par rapport aux interventions déclenchées par l'enseignant. En particulier, la visualisation du taux d'achèvement par exercice donne un aperçu précis de l'efficacité des interventions planifiées en montrant combien de *commits* sont réalisés par les étudiants après chaque type d'intervention. Un enseignant peut alors facilement prendre conscience de l'impact de ses interventions. Plus les interventions planifiées font partie de la conception de l'apprentissage, plus les informations obtenues sont riches. En effet, dans l'étude de cas n°2 où les évaluations par les pairs n'étaient pas explicitement prévues, il n'a pas été possible de mesurer l'impact des revues par les pairs qui étaient improvisées car elles n'étaient pas prises en compte dans l'outil comme des jalons.

D'autre part, les informations fournies par Git4school étant faciles à partager, Git4School peut aider à l'adoption de techniques, de ressources et/ou d'outils pédagogiques à fort impact à plus grande échelle dans une institution.

Identification des étudiants en difficulté

La visualisation du type de *commits* pour chaque élève permet d'identifier rapidement les élèves ayant le plus de difficultés. Par exemple, dans la figure 4, on peut voir que la quatrième barre est dominée par l'orange et le rouge, ce qui indique que l'élève correspondant fait rarement les exercices tout seul et, dans la plupart des cas, après la publication de la solution par l'enseignant. Lors de séances en face à face avec de petits groupes d'élèves, l'enseignant peut ne pas avoir besoin d'un tel outil pour identifier les élèves en difficulté. Cependant, Git4School donne rapidement des indicateurs pour identifier les élèves en difficulté avec lesquels l'enseignant peut décider d'interagir sans attendre qu'ils demandent de l'aide. Git4School permet ainsi aux enseignants de mieux répartir le temps passé avec les élèves sur la base d'indicateurs objectifs sur leurs besoins et a donc été utilisé comme aide pour réaliser des interventions non planifiées ciblant les élèves en difficulté.

Dans le contexte de l'enseignement hybride ou à distance, induit par les confinements liés à l'épidémie COVID19 par exemple, Git4School a été très utile pour identifier les élèves en difficulté en compensant en partie l'absence de signaux donnés par les élèves traditionnellement lors des sessions en face à face.

Information sur l'activité des étudiants en dehors des cours

La vue d'ensemble (Figure 3) affiche tous les *commits* effectués par les étudiants lors des sessions en face à face (fond bleu clair) et en dehors de la classe (fond blanc). Git4School permet ainsi d'identifier les étudiants qui poursuivent leur travail en dehors des sessions en face à face. Pour l'étude de cas 2, il s'agissait en fait d'un moyen de découvrir que moins de 5% des étudiants avaient l'habitude de travailler en dehors des sessions en face-à-face, même lorsqu'il leur était explicitement demandé de réaliser un travail à la maison. L'enseignant a questionné ce manque d'engagement des étudiants. Le paragraphe suivant décrit comment Git4School a permis d'identifier la cause la plus probable : un niveau de difficulté du travail demandé inapproprié au regard du niveau d'expertise des étudiants.

Qualification de la difficulté d'un exercice

La visualisation du taux d'achèvement par exercice (Figure 5) peut aider à identifier une situation anormale pour un exercice donné. Dans l'étude de cas 1, le seuil de déclenchement de la prochaine intervention prévue n'était pas atteint quand les élèves ne pouvaient pas effectuer l'exercice par eux-mêmes. Cette observation, pouvant être faite en temps réel par l'enseignant en consultant le tableau de bord, offre alors la possibilité de réaliser une remédiation sur les concepts qui sont visés dans l'exercice.

Si la visualisation du taux d'achèvement sur l'ensemble des exercices est largement dominée par la couleur rouge, cela signifie que les élèves n'ont réalisé les exercices qu'après la publication de la solution fournie par l'enseignant. Cette situation peut remettre en cause la conception pédagogique de l'ensemble des exercices proposés. En effet, dans l'étude de cas 2, comme les premiers exercices ont pris plus de temps que prévu, l'enseignant a anticipé la publication de la solution de ces exercices afin que les étudiants puissent consacrer plus de temps aux exercices suivants jugés plus importants par l'enseignant. Dans cette situation, Git4School fournit des indicateurs objectifs pour aider à identifier les ressources

pédagogiques inadaptées pour atteindre les objectifs d'apprentissage initiaux.

Comparaison de niveaux entre groupes d'étudiants

La deuxième étude de cas a impliqué deux groupes d'étudiants travaillant en parallèle sur les mêmes exercices. Assez rapidement, les visualisations sont apparues très différentes dans les deux groupes. Après les deux premières sessions, alors que le premier groupe montrait une distribution compacte des *commits* verts sur les quatre premiers exercices, le second groupe montrait une distribution plus large et inégale des *commits* verts jusqu'au septième exercice. Git4School est apparu comme un outil inattendu pour identifier les différences dans les progrès des étudiants d'un même groupe ainsi que l'hétérogénéité entre deux groupes inscrits au même cours. Ainsi, l'enseignant a pu donner du travail supplémentaire aux élèves les plus avancés afin de maintenir un rythme plus cohérent entre les deux groupes. D'autres choix auraient pu être faits (par exemple, mobiliser les étudiants avancés pour aider les autres), mais l'important ici est de constater que Git4School a permis la mise en œuvre d'interventions correctives imprévues déclenchées sur la base d'informations exactes.

Éthique et gestion des données personnelles

Git4School extrait des dépôts GitHub les données correspondant à la progression des étudiants, que ce soit lors de sessions en face à face ou en dehors des cours. Pour cela, les étudiants ont accepté les conditions d'utilisation de GitHub et de GitHub Classroom. Aucune condition d'utilisation supplémentaire n'est validée par les étudiants pour Git4School. Techniquement, aucune donnée n'est stockée sur le serveur qui héberge Git4School. Les données utilisées par le tableau de bord sont uniquement stockées localement pendant la période où l'enseignant utilise Git4School.

Limites

Actuellement, Git4School est uniquement compatible avec GitHub. Ce choix de conception n'est pas un problème pour les enseignants et les institutions souhaitant travailler avec cette plateforme populaire, mais il pourrait constituer un obstacle à l'adoption plus large de Git4School. En cas de frein avéré à son adoption, il sera possible d'adapter Git4School afin de le rendre compatible avec d'autres plateformes.

6. Conclusion

Nous avons présenté Git4school, un tableau de bord conçu pour soutenir les interventions des enseignants dans le cadre des cours de génie logiciel en fonction de l'activité des étudiants tracée via l'utilisation de dépôts Git. Une évaluation expérimentale nous a d'abord permis de fournir des réponses à (RQ1) en montrant l'adoption de Git et GitHub par les étudiants. Nous avons également montré en réponse à (RQ2) que Git4School peut aider les enseignants à prendre des décisions pour déclencher des interventions planifiées ou non planifiées. De plus, nous avons montré que Git4school a été utilisé pour évaluer l'impact des interventions planifiées et pour identifier les faiblesses dans la conception d'une situation d'apprentissage en permettant ainsi d'en cibler des améliorations.

Pendant les séances de la deuxième étude de cas, l'enseignant a montré à plusieurs reprises aux élèves la visualisation du taux d'achèvement par exercice afin de commenter la progression globale du groupe. Il n'a pas été possible de mesurer l'impact de cette utilisation particulière de Git4shool comme outil de sensibilisation des élèves, mais l'attention et la curiosité des élèves à l'égard de l'outil nous ont incités à réfléchir à la conception de visualisations dédiées aux élèves dans le cadre de telles activités. Ainsi, nous souhaiterions à l'avenir fournir aux élèves des visualisations spécifiques pour les aider à autoréguler leur apprentissage.

De nombreuses données ont été collectées au cours des expériences et d'autres viendront rapidement des futures utilisations de Git4school. Il sera bientôt possible d'effectuer une exploration statistique de certaines données afin de découvrir dans quelle mesure certains modèles de travail pourraient être corrélés avec les résultats d'apprentissage. Ce travail sur les données collectées est l'une de nos priorités pour améliorer la compréhension et l'évaluation des modèles d'apprentissage destinés aux cours de génie logiciel.

RÉFÉRENCES

Bakharia, A., Corrin, L., De Barba, P., Kennedy, G., Gašević, D., Mulder, R., Williams, D., Dawson, S. et Lockyer, L. (2016). A conceptual framework linking learning design with learning analytics. Dans *Proceeding of the International Conference on Learning Analytics & Knowledge (LAK)* (p. 329-338). ACM Press.

Bourque, P. et Fairley, R. E. (2014). SWEBOK: *Guide to the Software Engineering Body of Knowledge*. IEEE Computer Society. http://www.swebok.org/

Choquet, C., Delozanne, E. et Luengo, V. (2007). Numéro spécial : Analyse des traces d'utilisation dans les EIAH. *STICEF, 14.*

Emin, V., Pernin, J.-P. et Guéraud, V. (2011). Scénarisation pédagogique dirigée par les intentions. *STICEF, 18*(1), 195-227.

Gašević, D., Dawson, S. et Siemens, G. (2015). Let's not forget: Learning analytics are about learning. *TechTrends, 59,* 64–71.

Glassy, L. (2006). Using version control to observe student software development processes. *Journal of Computing Sciences in Colleges, 21,* 99–106.

ISCED (2012). *International standard classification of education: ISCED 2011*. UNESCO Institute for Statistics. http://uis.unesco.org/sites/default/files/documents/international-standard-classification-of-education-isced-2011-en.pdf

Jones, C. (2010). Using subversion as an aid in evaluating individuals working on a group coding project. *Journal of Computing Sciences in Colleges, 25,* 18–23.

Kay, J., Maisonneuve, N., Yacef, K. et Zaïane, O. (2006). Mining patterns of events in students' teamwork data. Dans J. Kay, N. Maisonneuve, K. Yacef et O. Zaïane (dir.), *In Educational Data Mining Workshop, held in conjunction with Intelligent Tutoring Systems (ITS)* (p. 45–52).

Liu, Y., Stroulia, E., Wong, K. et German, D. (2004). Using CVS Historical Information to Understand How Students Develop Software. Dans *Proceeding of the International Workshop on Mining Repositories (MSR)* (p. 32–36).

Lockyer, L., Heathcote, E. et Dawson, S. (2013). Informing pedagogical action: Aligning learning analytics with learning design. *American Behavioral Scientist, 57,* 1439–1459.

Mierle, K., Laven, K., Roweis, S. et Wilson, G. (2005). Mining student CVS repositories for performance indicators. *ACM SIGSOFT Software Engineering Notes, 30,* 1-5.

Mittal, M. et Sureka, A. (2014). Process Mining Software Repositories from Student Projects in an Undergraduate Software Engineering Course. Dans *Proceedings of the International Conference on Software Engineering (ICSE)* (p. 344–353). ACM Press.

Poncin, W., Serebrenik, A. et Van den Brand, M. (2011). Mining student capstone projects with FRASR and ProM. Dans *Proceedings of the ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)* (p. 87–96). ACM Press.

Poumay, M. (2014). Six leviers pour améliorer l'apprentissage des étudiants du supérieur. *Revue internationale de pédagogie du supérieur, 30*(1), 1–19.

Robles, G. et Gonzalez-Barahona, J. (2013, 3). Mining student repositories to gain learning analytics. An experience report. Dans *Proceedings of the IEEE Global Engineering Education Conference (EDUCON)*, IEEE (p. 1249-1254).

Schwendimann, B. A., Rodríguez-Triana, M. J., Vozniuk, A., Prieto, L. P., Boroujeni, M. S., Holzer, A. et Dillenbourg, P. (2017). Perceiving Learning at a Glance: A Systematic Literature Review of Learning Dashboard Research. *IEEE Transactions on Learning Technologies, 10,* 30-41.

Silvestre, F., Vidal, P. et Broisin, J. (2015). Reflexive learning, socio-cognitive conflict and peer-assessment to improve the quality of feedbacks in online tests. Dans G. Conole, T. Klobučar, C. Rensing, J. Konert et E. Lavoué (dir.), *Design for Teaching and Learning in a Networked World*, (p. 339–351). Springer.

Tchounikine, P. (2013). Clarifying design for orchestration: orchestration and orchestrable technology, scripting and conducting. *Computers & Education, 69*, 500–503.

Tissenbaum, M. et Slotta, J. (2019). Supporting classroom orchestration with real-time feedback: A role for teacher dashboards and real-time agents. *International Journal of Computer-Supported Collaborative Learning, 14*, 325–351.

Van Harmelen, M. et Workman, D. (2012). Analytics for learning and teaching. *CETIS Analytics Series, 1*, 1–40.

Wise, A. F. (2014). Designing pedagogical interventions to support student use of learning analytics. Dans *Proceedings of the International conference on learning analytics and knowledge (LAK)* (p. 203–211). ACM Press.