International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:14, No:10, 2020

Designing an Editorialization Environment for Repeatable Self-Correcting Exercises

M. Kobylanski, D. Buskulic, P.-H. Duron, D. Revuz, F. Ruggieri, E. Sandier, C. Tijus

Abstract**—**In order to design a cooperative e-learning platform, we observed teams of Teacher [T], Computer Scientist [CS] and exerciser's programmer-designer [ED] cooperating for the conception of a self-correcting exercise, but without the use of such a device in order to catch the kind of interactions a useful platform might provide. To do so, we first run a task analysis on how T, CS and ED should be cooperating in order to achieve, at best, the task of creating and implementing self-directed, self-paced, repeatable self-correcting exercises (RSE) in the context of open educational resources. The formalization of the whole process was based on the "objectives, activities and evaluations" theory of educational task analysis. Second, using the resulting frame as a "how-to-do it" guide, we run a series of three contrasted Hackathon of RSE-production to collect data about the cooperative process that could be later used to design the collaborative e-learning platform. Third, we used two complementary methods to collect, to code and to analyze the adequate survey data: the directional flow of interaction among T-CS-ED experts holding a functional role, and the Means-End Problem Solving analysis. Fourth, we listed the set of derived recommendations useful for the design of the exerciser as a cooperative e-learning platform. Final recommendations underline the necessity of building (i) an ecosystem that allows to sustain teams of T-CS-ED experts, (ii) a data safety platform although offering accessibility and open discussion about the production of exercises with their resources and (iii) a good architecture allowing the inheritance of parts of the coding of any exercise already in the data base as well as fast implementation of new kinds of exercises along with their associated learning activities.

Keywords**—**Distance open educational resources, pedagogical alignment, self-correcting exercises, teacher's involvement, team roles.

I. Introduction

LEARNER centered teaching that allows students to participate in the evaluation of their own work is a kind of regulated learning situations that provide them repetitive problem solving by having self-directed, self-paced, self-correcting exercises (RSE); so that learners can evaluate by themselves when they have accomplished the designed task [1] and how well they have reached the learning outcomes [2]. Cooperative e-learning platform hosting online, interactive, random, RSE in many different fields such as mathematics, chemistry, physics, biology… make it possible to favor learner centered teaching and students' acquisition of knowledge and know-how [3]. However, although the literature shows that properly used RSE have positive effects on the learning curve of students [4], one can notice some pitfalls: educational resources are rare and, when they do exist, it happens that there is either a lack of exercise-to-learning alignment when the exercises do not correspond to the objectives of the course, or there is a lack of exercise-to-evaluation alignment when students find how to succeed exercises without understanding and without learning.

In the context of Open Educational Resources, the issue of editorialization consists in creating a technical and human environment that allows the edition of exercises that meet as well technical as pedagogical criteria. While there are many platforms for creating exercises, the variety of types of exercises offered is generally limited and far from exhausting the possibilities offered by the web. These are often standard exercise formats, such as Multiple Choice Questions (*Socrative, Quizlet, Google classroom*). In addition, often these resources are not shareable, or difficult to share because they are difficult to reference. When they are referenceable, the identification of sources for curation purposes is not always assured, and when there is access to resources of self-corrected exercises, they cannot be changed.

There are platforms with predefined courses (Khan Academy, Aleks). In this case, the exercises can be very rich. However, articulation with classroom work and pedagogical alignment may be lacking, otherwise the pedagogical freedom is reduced: the exercises may correspond to objectives that are not those of the teacher, or to the national program but to the program of study of another country. Indeed, producing, referencing and sharing RSE is a difficult task that requires a good IT environment and numerous skills, related to discipline, pedagogy and programming, but also related to the design and implementation of activities involving these RSE and finally building a learner experience that is embedding the efficiency of the RSE objectives, activities and evaluations. Thus, the problem at hand is how to edit such exercises within an easy-to-use RSE platform and what the editorialization process ought to be.

Our research on this topic is about the editorialization process of the making of self-correcting exercises within an e-learning platform allowing Teachers [T], Computer Scientists [CS] and exerciser's programmer-designer [ED] to cooperate for this conception. Our current research on this topic is twofold. In the one hand, it is related to WIMS (Web Interactive Multipurpose Server) which is a collaborative, open source e-learning platform. Since its launch in 1997, WIMS has achieved the singular performance of proposing a tool allowing the production of self-correcting exercises, their sharing, and a prescription environment of these resources with indicators allowing students to follow their work and teachers to monitor the progress of the class as well as the one

Magdalena Kobylanski is with UPEM, France (e-mail: magdalena.kobylanski@u-pem.fr).

of each student (learning analytics) [3]. We have not met any other platform that allows to produce, share and modify RSE so easily. However the production of RSE in WIMS is made by individual teachers. The exercise writer must have strong disciplinary competencies, as well as pedagogical, coding and referencing ones. Moreover, in WIMS, once the RSEs are published the bug tracking relies on the RSE creator, for life-time. Thus, the community around WIMS might be interested by a more rationalized editorialization procedure that allows any teacher to participate to the creation of RSEs by defining efficient interactions between teachers, exercise's designer and computer scientists. On the other hand, the research results might provide guidelines about how to design an easy-to-use RSE platform with an editing procedure ranging from the training objective to the quality control of the exercises, making it possible to involve varied and specialized skills throughout the process. The purpose of this article is then to report the observations we made in order to infer the practical functions and corresponding tools that will satisfy the people wishing to set up this type of RSE platform, the existing ones or those to be designed and developed as well as to discuss and to formalize the editorialization process and procedure of this type of environment of production of resources.

A. Research Questions

We reasoned that a rational editorialization procedure of producing a RSE is the one that organizes the process of interactions between T, CS and ED, cooperating to the production of open educational resources (OER) managed by a community. This procedure can then be sustained by digital tools, workflow and processes, facilitating the production and rise to quality of self-replicated exercise-type learning resources.

We start from the needs for the training that is necessary to master certain types of learning: the so-called intended learning outcomes (ILO), a RSE being a response to the satisfaction of these needs: it allows targeting ILOs. We then listed a number of obstacles, difficulties and constraints a rational editorialization should overcome:

- exercises that are difficult to code
- resources that are difficult to reference
- learnings that are difficult to transfer from one context to another
- exercises that are difficult to prescribe or difficult to align with the course objectives
- students for whom it is difficult to diagnose the difficulty of success
- the difficulty of imagining and finding effective exercises and resources when existing exercises are not sufficient or do not meet the needs of students
- difficulty in characterizing the pedagogical effectiveness of a resource according to the context (among other available resources) and according to the type of student (for which student's profile).

Thus final recommendations should define: (i) an ecosystem that allows to sustain teams of experts in programming as a platform allowing data safety, accessibility and discussing about the production of resources, (ii) a good architecture allowing the inheritance of part of the code of an exercise and the fast implementation of new types of exercises and activities, (iii) an ecosystem that must sustain a community of experts sharing a common vision about the quality of pedagogical resources and seeking to produce it.

In addition, the editorialization' procedure of producing a RSE should be easy to use and teachers should gain some satisfactions for joining the workflow and processes cycle of production of exercises .

B. Introducing the Paper

In this paper, we report the observations of a Teacher [T], a Computer Scientist [CS] and an Exercise Designer [ED] cooperating for the conception of a self-correcting exercise and how this was modeled in order to design such a cooperative e-learning platform.

First, we define what are the “objectives, activities and evaluations” of such a task of creating and implementing a self-correcting exercise in the framework of the theory of educational task analysis: how T, CS and ED should cooperate.

Second, we run a series of three contrasted Hackathon of RSE-production to collect data about a cooperative process to be later used to design the collaborative e-learning platform.

The experiment in 3 sessions has had the objective of revealing production processes, in schematic form, for each Hackathon in order to analyze their evolution. It also helped to highlight the functions and strategies of teachers, computer scientists and of exerciser's designers.

Third, we used two complementary methods to collect, code and analyze the adequate survey data. The next step consists in listing the recommendations we derived from our observations and analysis in order to design an exerciser as a cooperative e-learning platform.

The final recommendations underline the fulfilments of the needs of a relevant architecture allowing the inheritance of parts of the code of an exercise and the fast implementation of new types of exercises and activities as well as the necessity of having a relevant ecosystem allowing discussion about the dimensions of data safety, accessibility in the production of resources as well as sustaining the programming teams engaged in the production of RSEs.

II. Theoretical Framework: How to Catch a Profitable Exercise

In order to avoid pitfalls related to compartmentalization (exercises that not related to others), of fragmentation (a given content being divided in parts that are used in unrelated exercises) and of unalignment of exercises and of ILO (exercises are not related to the targeted content), some authors [5], [6] stated that the successful learning object implementation should address three components: (i) an instructional design theory, (ii) a learning object taxonomy and (iii) a prescriptive linking material that connects instructional design theory to taxonomy, providing such guidance as “for this type of learning goal, use this type of learning object." While agreeing these pedagogical recommendations, we reasoned that the collective task of producing RSE might be based on sharing "objectives, activities and evaluations" dimensions as well as a common set of values and of criteria.

A. The "Objectives, Activities and Evaluations" Dimensions of the Designing of a Cooperative E-Learning Platform

Teachers (T), computers scientists (CS) and exercisers' designers (ED), - although participating in the shared goal of implementing exercises in an easy-to-use RSE platform -, do have their "own logic": the logic of T of final usability (the students of T), the logic of CS for computing the functionality allowing usability (the algorithmic coding of an exercise by CS that will be used by the students of T), the logic of DE for designing the interaction between usability and functionality (ED's composition of the input-output computer interface between the functionality that will be used to solve the exercises of the students of T). Moreover, as an important objective of the process is to produce resources that can be reused by other teachers and students, ED might ask T not being too much idiosyncratic; a constraint that CS should inherit in turn. The T-CS-DE dialoging necessary to overcome this task-to-function gap was theorized [7], [8] as a three levels hierarchy problem (user-Task, CS-function, ED-command) that can be solved at the command level. Following [7], [8], we reasoned that this T-CS-DE dialog might be applied not only to the *objectives* of implementing exercises, but also to the *activities* to reach these objectives and to the *evaluation* phase of measuring how much these objectives were reached. Thus, we derived the following rules about objective, activity and evaluation.

1. Any learning activity is a task-oriented activity done to reach a pedagogical objective. Learning outcomes of objectives have to be explicitly described.
2. Learning objectives are goals that require subsequent activities with prerequisites to be fulfilled. WHY and HOW such activities proceed to objectives achievement must be formalized. Learners should have the knowledge of the outcomes of the objectives in order to understand WHY and HOW such activities are to be undertaken.
3. Evaluation is about the fulfillment of the learning outcomes as defined by the objectives.
4. The evaluation about how much the objectives were fulfilled is independent of what has been done for, which is to say of the processes and procedures that were used through the activities.
5. Although an objective can be reached through diverse kinds of activities, understanding the evaluation results and explaining how to improve the learning process requires the knowledge of the prescribed ideal activity and how this was done as asserted by the specific learner (the observed activity and procedure).
6. The design of activities is considering that the quality of the student-RSE interaction will impact the evaluation process. Similarly, while doing the activities, learners should be aware that the way they solved activities will impact the automatic evaluation process that will detect errors and that learning by errors is also of benefit. Thus, for instance, instead of acting for a summative evaluation, the student would act for a formative one, finding more profitable to make errors instead of doing a "copy-paste" of the correct answering of a peer.

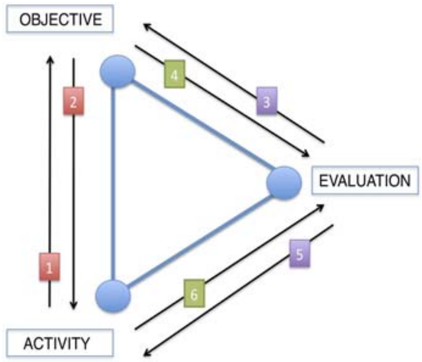

Fig. 1 Relations between objective, activity and evaluation

B. Modeling Objectives, Activities and Evaluations

Objectives: The team is having the goal to build an exercise to be implemented in the platform, which is the instrument they have to use while the exercise is the object of their task. This object, in our case the exercise, has to follow a flow of transformations: from scratch it has to become a self-directed, self-paced, self-correcting exercise (RSE) accessible and solved by learners using the same instrument-platform allowing the team to work together.

Activities to reach the targeted ILO through objectives with their goals and sub-goals are done by people (here a team made of a Teacher [T], a Computer Scientist [CS] and an exerciser's designer [ED]) using as instrument: a platform allowing students to solve on the screen of the used device (a smartphone, a tablet, a computer) the exercises of T that have been computed by CS; T and CS cooperating through ED that has the expertise of how learners will solve T's exercises in the CS' programed environment. The team is cooperating because any operation, from primitive actions to the highest level of obtaining from students the targeted learning outcomes needs the team consensus.

Instrument – Exerciser: a digital platform allowing a team made of T, CS and ED to cooperate in order to implement self-directed, self-paced, self-correcting exercises (RSE) to be solved by students.

Object – Exercise: An exercise corresponding to a task to be performed by students, with its components: the ILO, the title, the statements, procedure, interface… at successive states of transformation: E1 to E5.

- *E1- Definition of teaching objectives*: The exercise is in the head of the Teacher as a task to be solved by students in order for them to acquire the intended targeted learning outcomes (ITLO).
- *E2 - Definition of an activity aimed at acquiring and/or evaluating the acquisition of the targeted ILO*: the

exercise is an external representation, e.g. written on a paper by the Teacher.

- *E3 - Definition of the exercise algorithm*: The exercise is within a form filled by the teacher, the computer scientist and the exerciser designer. Dialogue between them as well as discussion with several teachers is profitable for E3 step and for the next states of transformation.
- *E4 – Coding*: The aim of this stage is the coding of the exercise done by a computer scientist in the programming language supported by the exerciser.
- *E5 – Publishing*: The exercise is the one that can be solved by students within the corresponding computer interface of the exerciser instrument.

Each member of the team has the following function-role (note that a given persons might fulfill more than one function/role).

- *T - Teacher - Pedagogical/Teaching*. Designing exercises to teach, train or evaluate students about their knowledge and know-how. Her goal ends when she got her exercises implemented on line through an exerciser. Her means is to fill out a sheet on targeted learning outcomes [TLO]. The teacher function is to transform E1 in E3.
- *CS - Computer Scientist* - Programming the exerciser in order to implement the computer with the exercise as described within the form with its TLO. Her function is to transform the form filled by the teacher in natural language into a computer language, coding the exercise through algorithms, commands and operations. The Computer scientist function is to transform E3 in E4.
- *ED - Exerciser's Designer* - the instrumental system, that is the computerized platform, should implement the exercises designed by the teacher to allow the students reaching the underlying TLO. The exerciser system function is to transform E4 into E5.

Evaluations: According to [9], evaluation is important because "various participants will have different ideas about what the program has to be and why and how it should work that way. Thus, creation of a program theory model serves to align the various stake holders into a single view". We incorporate the following part of Weir's set of six recommendations from engaging stakeholders about identifying the goals activities, resources, and context of the program and the measurement of success according to usefulness, feasibility, ethics, and accuracy, to gathering evidence with data, interviews and measuring outcomes measuring with the research design: comparisons against standards, statistical evidence, or expert review and dissemination activities.

In addition of having a team agreeing on objective, activity and evaluation, we argue that they should share a set of values and of quality criteria.

C. Sharing a Set of Values and Criteria of Exercise Quality

When having a group of T, CS and of ED, they can start defining values and criteria of exercise's quality. This is to be somewhat the ideal and exemplarity of the eLearning platform. Defining what are the underlying properties of learning objects that can be used as indicators of a satisfying eLearning content was made for MOOCs [10].

Among values to share among people developing eLearning platform and exercises to implement, we promote those of [11] that are both technological and pedagogical: "(i) the right to make, own, and control copies of the content (e.g., download, duplicate, store, and manage), (ii) to use the content in a wide range of ways (e.g., in a class, in a study group, on a website, in a video), (iii) to adapt, adjust, modify, or alter the content itself (e.g., translate the content into another language), (iv) to combine the original or revised content with other material to create something new (e.g., incorporate the content into a mashup), (v) to share copies of the original content, your revisions, or your remixes with others (e.g., give a copy of the content to a friend)". We also promote as [10] the properties of a eLearning content that can be used as indicators to evaluate how much a RSE is satisfying the students and teacher needs:

- Technological related properties: 1) the availability of the exercise as resource, 2) its permanence, 3) its accessibility, 4) its presence in the inventory of available resources, 5) its openness to other resources,
- Pedagogical related properties: 6) the completeness and autonomy of the exercise as a resource, 7) its interdisciplinarity, universality and timelessness,
- Educational related properties: 8) its cognitive and pedagogical treatment, 9) its ergonomic and educational efficiency, 10) its educational effectiveness.

III. Methodology: How to Proceed to Design a Cooperative E-Learning Platform

Because our goal was to find the editorialization functions a (T – CS - ED) cooperative e-learning platform might have to facilitate the easyness of implementing RSE. Our experimentation was to observe teams T, CS, ED cooperating to create, formalize, code and display RSE in natural environments that allow us to infer their needs and the functions to fulfill them. In order to observe how T, CS and ED would cooperate to build RSEs related to ILO as if they were going to use a next eLearning Platform, we used two methods. First method was to collect all of the interactions between the T-CS-ED participants all along the E1-to-E5 process of transforming the ITLO-based exercise project to the eLearning platform implemented exercise to be solved by students.

In the field of interaction, course design, and evaluation in distance education [12], [13], there is a distinction between learner-teacher, learner content, and learner-learner [14]. However, it is of importance to consider the interaction between the teacher that is a content expert and an instructional expert about how to teach this content and the computer scientist that as the functional expertise of how to implement this content in digital, at distance, cloud computing devices. An exerciser designer (ED), - the expert that knows students and teacher needs as well as the technical constraints and possibilities, can be facilitating this interaction because s/he can model the students' behavior.

In order to observe the T, CS and ED interactions, we used the framework of situated cognition [15] stating that activities during which knowledge is constructed are components of that knowledge [16]: (i) Identify the objectives of instruction, (ii) Select the useful learning experiences, (iii) Organize the learning experiences in the best possible manner, and (iv) Evaluate learning. More precisely, the method was to capture the flow of interactions between T, CS and ED participants in the framework of the Means-End analysis of implementing an RSE [17]. Appendix 2 displays an example of how data of interactions were collected.

Means-end analysis is one of the methods of analyzing problem solving processes. It stems from the Herbert Simon theory who considers the situation in which the solution will fit as a problem space; namely a space of states and of states transitions including characteristic states (initial state, final state), legal transitions (the actions that make it possible to pass from one state to another of the states of the problem structure) and dead-ends which are states from which no transition is possible to move to another state except to go back to the preceding state by undoing what had just been done [18]. Transforming the exercise from E1-to-E5 is this kind of problem structure.

The individual or collective difficulty in solving the problem stems from two types of difficulties: (i) the difficulty inherent to the objective constraints of the problem for which we must find/infer/discover the legal operator who allows to change state, but especially in an effective way; which means finding the shortest way to reach the goal [18] and (ii) an additional difficulty that comes from the problem solver who does not intend to do any actions although allowed or to achieve states s/he thinks do not exist or are impossible to reach [19]. Thus, the resolution difficulties of implementing an exercise might come from:

- *D1* - from the evaluation of the gap to the goal: the people compare the states that they can reach with the state-goal and select the state which appears to them the most similar to this state-goal, wanting to take a shortcut while a detour is necessary,
- D2 - an inadequate representation of the problem space:
  - D2.1 - people imagine states that do not exist and try to reach them
  - D2.2 - people do not realize that they are in an objective stalemate
  - D2.3 - people are convinced that they are in an impasse and are ready to give up,

The means-end analysis thus makes it possible to identify and correct the collective resolution difficulties that are of one of the four types above (D1, D2.1, D2.2 and D 2.3); here to perform an exercise that will be implemented in a platform.

IV. Method

A. Participants

Participants were from 5 universities, being either (i) teachers that come from disciplines such as mathematics, physics, chemistry and computer science, (ii) computer scientists or (iii) EDs that were experts about the functioning of eLearning Platform, coaching teachers and computer scientists. They agreed to play this attributed function in the 3 following sessions, named Hackathon.

- Session 1 - First Hackathon grouped 18 participants: 11 teachers (6 in chemistry, 2 in physics, 3 in Mathematics), 3 computer scientists and 5 EDs.
- Session 2 - Second Hackathon grouped 9 participants that participated in session 1, 6 were new. For session 2, they distributed in three teams: 1) the chemistry team (2 teachers and 1 ED), 2) the info team (3 teachers and 4 computer scientists), 3) the physics team (3 teachers with a computer scientist and an ED).
- Session 3 - Third Hackathon grouped 12 participants working individually as teachers (3), as teachers being also Computer Scientists (7) or as teachers being also EDs (2).

B. Materials

- *Session Material 1* - A room divided into 3 types of space according to the function of the participants: (i) 3 spaces for teachers (1 in math, 1 in physics, 2 in chemistry), (ii) 1 space for computer scientists and (iii) 1 space for EDs.
- *Session Equipment 2* - Two rooms divided into 3 thematic areas: (i) computer science, (2) physics, (3) chemistry.
- *Session Material 3* - Individual access to an online collaborative platform with hotlines and appointments, videos explaining the process, a link to cards to fill, and the use of Github, a global platform for developer collaboration, helping developers to build and design software.

C. Procedure

Rationales of observations: The 3 sessions were planned in order to test the effects of three contrasted conditions:

1) a cooperation between T, CS and ED cooperation: *session 1* - T, CS and ED were in different locations into a room, in such a way that collaboration necessitates to move from one space to another space: first teachers do their work, then cooperate with CS/ED.
2) a cooperation within T, CS and ED cooperation: *session 2* - T, CS and ED were in a same location, in such a way they could cooperate without necessity to move from one space to another space.
3) an individual participation, using a platform to distance collaboration or cooperation: *session 3* – Most of the participants are able to fulfill 2 or more of the functions: thus they can collaborate or cooperate with others as any of the 7 topics of interaction: (i) T/T, (ii) T/CS, (iii) T/ED, (iv) T/CS/ED, (v) CS/CS, (vi) CS/ED, (vii) ED/ED.

Whatever the session (1, 2 or 3), teachers were asked to define the ILO of the exercise, to fill in an application form for implementation of an exercise. This application form was specifying its educational objectives, describing its contents, by listing and specifying the variables and by indicating how the response and the feedbacks to be delivered are determined.

- *Session 1 procedure* - Grouped together, according to the discipline, the teachers had to complete together the application form, which was then to be brought to an ED, located in another place, who could then ask for any clarification by writing on the form. This validated form was then transmitted to a computer scientist, located in a third space, for the coding of the exercise. S/he could ask for clarification, either to the ED, or directly to the teachers who wrote the form, or to other teachers of this discipline.

- *Session 2 procedure* - Grouped by team composed of teachers, computer scientists and exercise designers, the participants work together to produce the application form of each exercise. The exercises are then coded.
- *Session 3 procedure* - The participants work remotely individually to produce the exercise application form using the Github platform to capture the exercise information, but also to collaborate and cooperate. A spreadsheet allowed participants to register for face-to-face coaching sessions with a computer scientist and/or an exercise designer. The resulting form then serves as a support for discussion, moderation, attribution, and validation via the Github Issues.

At this stage, the data collected for the present study are the application forms of exercises at the E1-to-E5 steps of the exercise transformation, from the ILO to the ready-to-implement exercises. Other data are the recording of verbal interactions, photos, screenshots, and exchanges between participants at the time and place of their interaction (see appendix 2).

D. Means-End Analysis Coding

To understand the problem solving process of creating and implementing an exercise in a dedicated platform, we observed the realization of possible goal and sub-goal states. These states can be represented graphically or described by nodes that represent any possible state between two actions and each link as a possible action between states, done by a human or by a technical system (GITHUB, for example), (see Appendix 2). The following tree describes the main goal and subgoals of the cooperative task.

1 – To have the Exercise implemented in the Exerciser by T
1.1 – Having the form filled with the exercise description
1.1.1 Having the empty form to fill
- Existence of an exercise form
- Having the template
1.1.2 Having the exercise description to fill the form
1.1.3 Having the form filled with the description
1.2 – To have the form accepted by ED
1.2.1 Form reception by ED
1.2.2 Form evaluation by ED
1.2.3 Feasibility verification by EC
1.3 – To have the form content coded by CS
1.3.1 Exercise taken in charge by CS
1.3.2 CS notifies T about her/his support
- Coding of the exercise
- Notification to T about the coding
- Finalization of the coding of the exercise
1.4 – To have T satisfied by the implemented exercise

V. Results

A. The Flow of T, CS and ED Interactions

Session 1 - This first session did not involve any intermediate digital production. The observation is that of a high number of exchanges between T and ED to transcribe a pedagogical need into an IT need. These observations attest to an important work of adequacy between T and CS; ED being a translator/mediator between them. The verbal communications observed between CS and ED are aimed at improving knowledge in terms of feasibility to return to T the best possible advices. Once this work is done, the "application form" arrives on the coder table. However, a simple reading of the document is not generally sufficient to understand the T's needs. We have thus observed many verbal exchanges from CS to T to overcome the information deficit of the "application form" and reach an exercise. This process, far from the expectations of the designers, has nonetheless enabled a continuous improvement of the "application form". As the process was repeated over the three days of session duration, there was a reduction in oral interaction flows.

In summary, there is a heaviness in the collaborative work where the stages, from E1-to-E5, follow one another with difficulty of transitions between the successive states of the progress of solving the problem.

Session 2 - In this hackathon of the second session, participants of different functions were assigned to same team working all along the session. Because they fill together the application form of the exercise, the document to be completed by the teacher with the "comment" part allows the discussion between T, ED and CS. The main parts of the document to be completed initiate a "thread", named Issues, between the three parties. On the Github software, this "thread" allows the ED to classify the T's requests via a keyword system and the CSs to query T to access details that s/he did not find in the document to be completed. It follows that the number of questions of ED and CS were thus much fewer in the second session than in the first; showing thus a greater fluidity and a greater ease of transition from E1 to E5 which are the different states of the realization of the exercise. It is also the issues that allowed the assignment of the exercise to a CS and the easiness of the production the exercise once it coded. In addition, in session 2, T is satisfied more quickly with the resulting implemented exercise. This process can be explained by the fact that the possible blocking points were found and solved upstream of the process.

Session 3 - The Hackathon's goal was to test an editorialization process entirely remotely, but based on the successes and mistakes of the first two sessions. The transition to digital during this third session allowed participants to interact with asynchronous flows and, because a participant usually fulfilled several functions, to get very varied and profound exchanges, not only related to a function (e.g. DS with DS), or between functions (T with CS), but especially in a more efficient way for the making of RSEs, related to the links and relationships between functions (e.g., T/DE/CS).

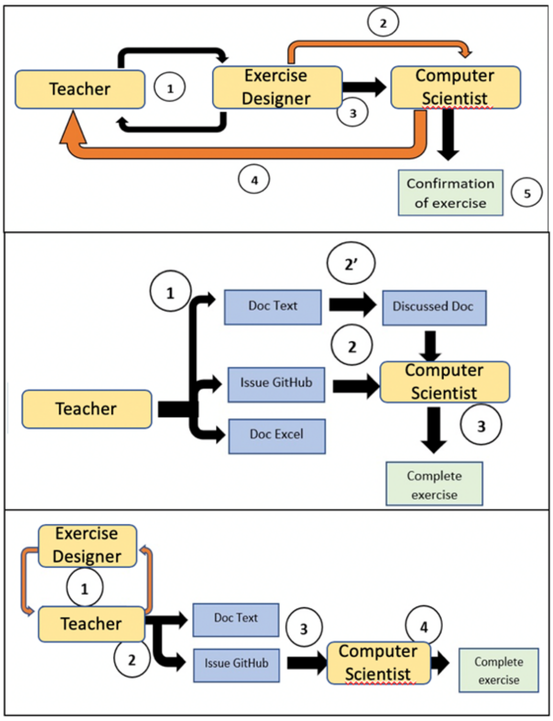

Fig. 2 Hackathon's verbal and instrumented interaction flow for session 1 (Top panel), for session 2 (Middle panel) and for session 3 (Bottom panel). The numbered circles show the main observed steps specific to each session. Yellow boxes correspond to the participant functions (Teacher, Computer Scientist and ED) that intervene in the RSE making process. Blue boxes correspond to the intermediate digital productions of the process. Green boxes correspond to the RSE final productions. Black arrows refer to non-verbal interactions between participants that lead to a production. Orange arrows refer to a verbal interaction, unwanted by the ED for the first session but encouraged by them for the third session, allowing understanding inter-functional specific topics

B. Means-End Analysis

In the initial state of the making of a RSE, we have T or a group of Ts who propose an exercise in a given discipline. S/he conceptualizes her/his pedagogical aims through ILO and formalizes his request in an application form by writing it according to precise criteria. This document is then propagated to a CS and/or an ED that receives it and checks it for compliance to respond to the request and start the program's informatic writing; the target state being the validation and publication of the exercise at the end of the editorial chain. In this collaborative development chain, narrowing the gap between a current state of the exercise development and the production of the exercise consists of going through the constraints of the system to move towards the final goal state (giving satisfaction to T by creating and validating the exercise); this is done passing through successive mandatory states. In other words, T must find the optimal path that links the states to the validation of his exercise, CS must write the code of the exercise and ED must see if the request made is consistent and feasible.

The choice of actions that make sense in the elaborative space of the exercise is usually guided by basic heuristics. These rules of action are empirical, simple, practical and especially fast to implement in situation by facilitating the discovery of solution highlighted by the means-end analysis of the process. It shows the actions and decisions taken at each stage (at the goal and sub-goals levels) of the exercise construction process. However, all of these decisions do not always guarantee their validity as a solution towards the ultimate goal, but they allow to restrict the number of relevant paths to explore to reach the goal state.

In solving design problems, such as the construction of CSR, design support systems must be able to handle a large number of constraints, with the aim of relieving human operator of difficulties.

On the mean-end analysis of the three sessions, we observed that the learning of participants intervenes at the moment when T is expressed by the use of optimum path between the request and the validation of an exercise and that it used the operators put at his disposal to speed up the execution of the resolution processes for this task. This is possible in two of the three sessions: 1) when the team is made up of participants of the 3 functions (T/CS/ED in session 2) or when individually working participants perform 2 or 3 functions and can interact with others thanks to the form that allows collaboration and cooperation. Thanks to this experience, he anticipates the obstacles and avoids the constraints of human operators, relying on the design support systems to formulate and formalize her/his request correctly upstream of the process.

The mean-end analysis of the conceptual resolution also shows the declarative knowledge (knowledge of the way to go) and the procedural knowledge (preferential use of this or that operator) to acquire for T to use the implementation of a typical solution in the elaborative process of an exercise. Indeed, each decision-making has an effect and a cost in time on the entire editorial chain.

VI. Discussion

According to ILO theory, quality of teaching and improving student successes are guaranteed by aligning objectives, activities and evaluations. The editorialization process is a way to link objectives, activities and evaluations in order to get RSE that are profitable to students.

For creating a given exercise with RSE design aids, the means-end analysis task for constructing an exercise in the editorial chain is to discriminate the optimal path from other less relevant solutions.

We described this workflow of creating an exercise involving participants who have different functions (T/CS/ED), by methodically breaking them all its processes into goals and sub-goals as well as the constraints between participants and devices that make up the chain, and also by describing the operators in terms of the rules of action that bind them. This is done starting from the formulation of a

International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:14, No:10, 2020

request, the propagation of this request in the editorial chain, the backtracking and the satisfaction of the final goal by a terminated and validated exercise.

Based on the blocking points we have identified, the strategies for circumventing these constraints, but also based on the explicit requests of the group of exercise designers and associated platforms; which are the spin-off of the results of our work, we list a number of recommendations:

- Transcribe the educational need into IT needs
- Fulfill needs for individuals on an exercise
- For a team of teachers (local or distributed teams), fulfill the need for training with examples, of procedures
- Find the right dimension/complexity, at the design level, of the structure of the application form (neither too much nor too little)
- Reusability of the application form for the reusability of the exercise
- Have a plug editor to create visual and logical components of the application form
- Need pedagogical interactions between people of the same discipline and with platform ED specialists
- Need a tool to manage the workflow that expresses the need of exercises platform
- Need a structured observation device (automatic collection of data)
- Adapt the platform and for that a reassessment of the needs towards deep coders
- Make it possible to create new kinds of exercises
- Have different levels of coding

In addition, we observed that

- For tasks [E1] - [E3], a discussion among some Ts was profitable.
- The coach ED can also help T in tasks E1 to E3 when he actually acts as a "platform expert" and also in training for new T's,
- ED makes a first sort as in emergencies (identify exercises easy to code, qualify the difficulty of coding and therefore the expertise needed to code it).
- The discussion on the application form (request for additional explanation from CS to understand) is rarely done with the ED, but rather between T and CS, or sometimes with a disciplinary expert.
- [E4] to [E5] should be done in agile mode (between CS and T)

VII. Conclusion

An important lesson of the Hackathons was that the computer tool is still very far from allowing a fast and effective coding for teachers: it lacks many functions as well as preprogrammed tools. In this sense, our study pushes to evaluate the computer tool with:

- the contribution of this study to the development of the platform and the network of contributors
- the strategies to reduce the gap between demand and creation of the exercise
- the contribution of the usefulness of application form: it allows a CS to produce exercises. However this editorial choice gives extra work to ED. For instance, it takes half a day of discussion with the teacher to translate his need into the application form, to transmission of the form to the engineer. If s/he has questions, the ED coach answers them. Then, once the exercise coded, there is the validation step. These choices make it possible to integrate CS with no disciplinary skills.

Appendix

Appendix 1

WIMS is a collaborative, open source e-learning platform hosting online interactive exercises in many different fields such as mathematics, chemistry, physics, and biology among others. It has been created in 1997 by Xiao Gang [3]. It is used mostly in France and mostly in mathematics in high school or during the first years in university. It provides a language to program and to produce self-correcting exercises as well as a repository to share them. It also provides a class environment to organize the resources for students of specific needs. The WIMS class is equipped with a rich range of tools including learning analytics.

Appendix 2

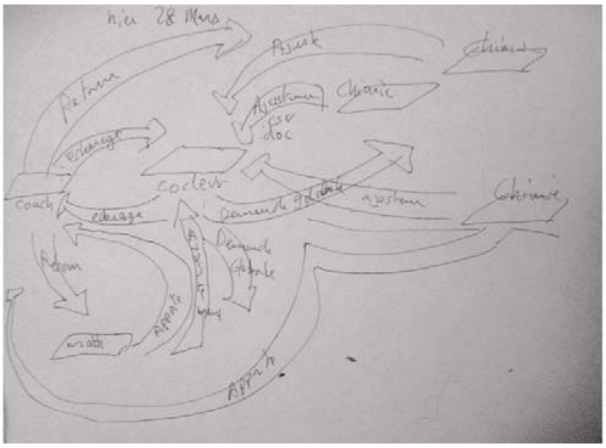

Fig. 3 Collection of the of T, CS and ED interactions

Acknowledgment

The authors thank all the Hackathons' participants: the one that endorsed the teacher's role as well as the one that endorsed the computer scientist's role, and of course the one that played the exercise designer's role.

References

[1] D. Callison, "Inquiry in Science and Math. The Evolution of Inquiry: Controlled, Guided, Modeled, and Free: Controlled, Guided, Modeled, and Free," *SLM Hot Topics, ABC-CLIO*, 2015, p. 301.

[2] G. Karlsson, C. Johannesson, J. Thorbiörnson, and M. Hellström, "Net Based Examination: Small Group Tutoring, Home Assignments, and Large Group Automatic and Peer Assessment," *International Journal of Emerging Technologies in Learning*, 2007, 2(3), pp. 1-5.

[3] M. Kobylanski, "WIMS: Innovative Pedagogy with 21 Year Old Interactive Exercise Software," *In Technology in Mathematics Teaching* Springer, Cham, 2019, pp. 123-144.

[4] C. B. Hodges, "Designing to motivate: Motivational techniques to incorporate in e-learning experiences," *The Journal of Interactive Online*

Learning, 2004, 2(3), pp. 1-7. [W.-K. Chen, *Linear Networks and Systems* (Book style).] [Belmont, CA: Wadsworth, 1993, pp. 123–135.]

[5] D. A. Wiley, “Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy,” *The instructional use of learning objects*, 2000, 2830(435), pp. 1-35.

[6] J. J. Van Merriënboer, and P. A. Kirschner, “Ten steps to complex learning: A systematic approach to four-component instructional design,” *Routledge*, 2017.

[7] J.-F. Richard, “Logique du fonctionnement et logique de l'utilisation (Rapport de recherche n° 202),” in F. Le Chesnay, “INRIA,” 1983.

[8] D. A. Norman, “Some observation on mental models,” in D. Gentner and A. L. Stevens (Edit.), “Mental models,” *Hillsdale (Nj), Erlbaum*, 1983, pp. 7-14.

[9] C. R. Weir, “Models, Theories, and Research for Program Evaluation,” In N. De Ramona, and N. Staggers (Eds), “Health Informatics-E-Book: An Interprofessional Approach,” 2016, pp. 60-74.

[10] L. Lachaud, G. Erjavec, L. Lellouche, and C. Tijus (2018). Mise en place d’un référentiel pour le MOOC exemplaire. *Rapport interne Projet Eiffel_a*. France Université Numérique.

[11] D. Wiley, “The MOOC misstep and the open education infrastructure,” In MOOCs and Open Education around the world, *Routledge*, 2015, pp. 3-11.

[12] C. Vrasidas, “Constructivism versus objectivism: Implications for interaction, course design, and evaluation in distance education,” *International Journal of Educational Telecommunications*, 2000, 6(4), pp. 339-362.

[13] C. Vrasidas, and M. Zembylas, “Online professional development: Lessons from the field,” *Education + Training*, 2004, 46(6/7), pp. 326-334.

[14] J. Moore, “Is higher education ready for transformative learning? A question explored in the study of sustainability,” *Journal of transformative education*, 2005, 3(1), pp. 76-91.

[15] J. S. Brown, A. Collins, and P. Duguid, “Situated cognition and the culture of learning,” *Educational researcher*, 1989, 18(1), pp. 32-42.

[16] R. W. Tyler, “Basic principles of curriculum and instruction,” In Curriculum Studies Reader E2, *Routledge*, 2013, pp. 60-68.

[17] A. Newell, and H. A. Simon, “Human problem solving,” Vol. 104, No. 9, 1972, Englewood Cliffs, NJ: Prentice-Hall.

[18] Y. Anzai, and H. A. Simon, “The theory of learning by doing,” *Psychological review*, 1979, 86(2), p. 124.

[19] J. F. Richard, S. Poitrenaud, and C. Tijus, “Problem-solving restructuration: Elimination implicit constraints,” *Cognitive Science*, 1993, 17(4), pp. 497-529.