Method for Joining Information and Adapting Content from Gamified Systems and Serious Games in Organizations

Mathieu Guinebert[1][a], Joël Fabiani[1][b], Melaine Cherdieu[1][c], Pierre Holat[1,2][d] and Charlie Grosman[1][e]

[1]*FI Group, 14 Terrasse Bellini, 92800 Puteaux, France*

[2]*Laboratoire d'Informatique de Paris Nord, Université Sorbonne Paris Nord, 93430 Villetaneuse, France*

Keywords: Serious Games, Gamification, Design Method, Scenario Formalization Model, Adaptive Learning.

Abstract: In some work fields, the number of various knowledge and skills one must master can be tremendous. Therefore, we decided to make work and training more rewarding and motivating. The skills and knowledge mustered by the employees in both those situations are the same, but the systems responsible for the tracking and the adaptation of the content are not. Therefore, our contribution is twofold. First, a system that centralizes the learner's game, learning and professional profiles and provides the other systems connected to it with the necessary information to adapt their content thanks to various modules. Secondly, a generic model that should be respected by any system connecting to our first element. We argue that it is necessary to use both our method and model to be able to fully exploit the information provided by our system. We tested our model and method on three different implementations but could not measure the impact of said implementations on our learners.

1 INTRODUCTION

"Evidence from the field of labour economics suggests a positive relationship between training and firm productivity" (Bryan, 2006). Moreover, as it is shown by the literature (Roussel, 2000), great motivation implies a greater implication and a greater efficiency in the given tasks and activities. Research on gamification and serious games relies directly on those principles. Indeed, they use game elements as a motivational motor (Alsawaier, 2018).

With this concept as our basis, we decided to put in place several systems in our company. Those systems are dedicated, on the one hand, to making the employee's training more playful through learning games and, on the other hand, to the gamification of their everyday work. However, as indicated by (Dale, 2014), the efficiency of such methods in the case of companies is not guaranteed. The use of a design method allows us to limit the inefficiency risks of those systems (Kappen & Nacke, 2013). Given the fact we wish to put in place various educational systems targeting the same skills (in a simulated context, and in a real one), we seek to create and implement a complex system that would allow those educational systems to be joined around their player-learner profiles, knowledge models, and skill models (both the pedagogical and playful ones). We also aim for this junction to be made around as their game logs and the equivalences between "professional" and "pedagogical" skills. This complex system, which we named "Joint System" (JS), is destined to be modular. The JS itself needs, also, a design method. Besides the tracking of the learners and its skills, our system is aimed at the increase of the playfulness of everyday work and training. It focuses itself, on the one hand, on maintaining the learner's motivation through the use of his/her logs to generate adapted playful content, and, on the other hand, on adapting the pedagogical content provided to the learner. We

[a] https://orcid.org/0000-0002-1777-6708
[b] https://orcid.org/0000-0002-9880-663X
[c] https://orcid.org/0000-0003-1368-4053
[d] https://orcid.org/0000-0003-4972-8679
[e] https://orcid.org/0000-0001-7100-121X

Guinebert, M., Fabiani, J., Cherdieu, M., Holat, P. and Grosman, C.
Method for Joining Information and Adapting Content from Gamified Systems and Serious Games in Organizations.
DOI: 10.5220/0010997500003182
In *Proceedings of the 14th International Conference on Computer Supported Education (CSEDU 2022) - Volume 2*, pages 338-350
ISBN: 978-989-758-562-3; ISSN: 2184-5026

intend for the JS to allow the future addition of modules such as ITS, authoring tools, CMS, etc. as needed.

Moreover, the JS must be able to differentiate skills issued from training and skills issued from professional tasks carried out by the employees. Therefore, the design method must consider the design and implementation of both our educational systems and of the adaptive modular system making the junction between them.

To guarantee the optimal tracking of our player-learner profiles, we decided to use exclusively computer-mediated solutions for us to complete the existing training. Those choices also seem quite relevant regarding the increase in remote working caused directly by Covid-19.

Our contribution can be summarized to three main components. Firstly, a design method for educational and playful systems taking part in our JS. Secondly, a generic scenario formalization model that can be used to describe both playful and pedagogical scenarios, and, thirdly, how to link said scenarios to the JS.

2 WHY A METHOD AND A MODEL

Some researchers present gamification and serious games as synonymous inside of their work (Caponetto et al., 2014), while others make the differentiation (Landers, 2014; Deterding et al., 2013). In our context, we decided to differentiate them. We place the difference between the nature of the gamified task. In this paper, a gamified system refers to systems in which real tasks are being gamified. Be it with or without game elements, the tasks wouldn't be any different and the consequences on the results of the employee won't change on a professional scale. On the other hand, serious games will refer to any systems constructed with a "serious" intent and in which the tasks are both playful and simulated. Therefore, any mistakes made in a gamified system have "real" consequences as opposed to mistakes made in a serious game.

2.1 The Need for a Method

The literature gives answer elements to our Research Question n° 0 (RQ0) "How do we guarantee the efficiency of gamified systems and learning games in a company?". Indeed, as indicated by (Kappen & Nacke, 2013), the efficiency of a gamified solution is directly dependent of its design. However, as (Nacke & Deterding, 2017) suggest, the gamification and everything linked to it has yet to reach full maturity. Moreover, the recent literature on serious games design method also translates a need for it. For example, we could quote the work of (Avila-Pesantez et al., 2019), who reminds us of this need in their literature review before presenting their own method.

Therefore, those observations lead us to our RQ1 (directly obtained from our RQ0): "Which approach is needed to guarantee both the efficiency and the relevance of a complex system composed of several serious games and gamified systems?"

Various leads can be found in the different approaches available in the literature to help the design of gamified systems and serious games. We mostly focused ourselves on four of them that decomposed their method into phases. Of course, those four methods are not the only ones to do so but we had to narrow down our choice to a manageable subset. Two of those four methods are directly focused on gamified systems: GOAL (Garcia et al., 2017) and (Morschheuser et al., 2017). The two others are centered around learning games: "the 6 facets" (Marne et al., 2012) and (Avila-Pesantez et al., 2019).

Those four methods do not always agree on the workflow. Our two learning games methods tend to give far more freedom on the matter (in particular (Marne et al., 2012)). There is also a lack of consensus on the very nature and number of the phases composing the method. Beyond specific consideration like the obvious lack of pedagogical objectives in the gamified systems methods, we can find several common points such as the development and evaluation phases.

However, none of those methods could satisfy us fully. Indeed, we emit the Hypothesis n°1 (H1) that to guarantee an optimal efficiency for our various systems destined to be connected to our JS, they need to be designed with the intent of being connected to said system and its various modules. Therefore, in order to verify our hypothesis, we need a design method taking into account the specificities of our JS that would allow for the design of both serious games and gamified systems. Moreover, we also emit the hypothesis H2 that it is possible to reach equivalences between pedagogical and "professional" objectives in such a way that "real" and simulated results could be used freely by any systems connected to our modular one. None of the methods and approaches that we could find in the literature seemed to consider both those hypotheses. Thus, explaining why, we had to create our own.

2.2 The Need for a Model

(Liu et al., 2017) reminds us that the benefits of an adaptive system on learning are fully admitted but needs a particular attention to its design. Moreover, as indicated by (Peng et al., 2019), the apparition of new technologies, most notably in the domain of Big Data and Data Analysis, incites us to construct new forms of learning using those information and technologies to better adapt to the learner.

Our JS allows us to link numerous concepts and identical skills but implemented and evaluated by different systems. Those concepts and skills can be of pedagogical nature, of course, but also of gamified nature. When a learner fails a gamified task, are we sure that he/she failed because of a lack of pedagogical skill? Or could he/she have failed because of gaming aspects? In this context, the possibilities offered by our joint system on the modeling of the learner has led us to consider both the playful and pedagogical adaptation of the content in our project.

As is indicated by the frequently cited Flow Theory (Csikszentmihalyi, 2000), it is important to adapt the difficulty of the task to the learner's skill level in order for his/her motivation not to plummet. Our player-learners are evolving in systems that can link both playful and pedagogical aspects. Therefore, the adaptation of the content can't rely on a unique Flow curb, but should rely on at least two curbs, one for the pedagogical aspects and one for the game aspects.

Thus, our hypothesis H3 in the scope of this project is that, in the context of serious games and gamified systems, the scenario to which the learners are exposed must be both adapted from a pedagogical point of view and from a playful point of view to ensure the learner's optimal motivation. Therefore, we seek to answer the following RQ2 "Which formalism or model to adopt in order to ease the adaptation and differentiation of playful and pedagogical scenarios?".

Regarding pedagogical models, IMS-LD (Hummel et al., 2004) is still today regularly cited (Ouadoud et al., 2018; El Moudden & Khaldi, 2018). IMS-LD address every problematic linked to the modeling, the design, and the organization of a system's pedagogical content. The Pleiades method (Villiot-Leclerq, 2007) is another interesting approach for the modeling of pedagogical scenarios. However, none of those two methods has been truthfully conceived to consider playful elements.

More recent approach such as "MoPPLiq" (Marne et al., 2013) or "Multiplayer Learning Game Ontology" (MPLGO) (Guinebert et al., 2017) makes the link between pedagogical and playful elements. Yet they are still not perfect for our context. In the case of MoPPliq for example, every activity sequence available to the learners need to be defined directly in the model. An adaptation using this model would thus be limited to the links defined in the scenario.

In MPLGO, the precedencies between the activities are flexible and are determined by the game resources produced and consumed by the players. However, the knowledge and skills are not considered in the construction of those precedencies. An adaptation based on MPLGO would only rely on playful elements which does not answer our problematic.

One of the closest answers we could find for our needs toward a model considering H3 was the model and methods proposed by (Marfisi-Schottman, 2012) which infers its pedagogical structure directly from IMS-LD and differentiate the game scenario. This model and method have been designed to help the communication between the various individual working on the Learning Game and seems to tackle the game scenario mostly on a narrative scale. Every detail of the most atomic component, the screen, which involves the interactions with the Learning Game itself is left to the screen designer.

The models associated with the gamified such as the GOAL ontology (Garcia et al., 2017) also fails to satisfy our needs. The pedagogical aspect is, for obvious reasons, often nonexistent in said models.

We failed to find a method or model in the literature that would satisfy our needs to adapt a scenario on both its playful and pedagogical aspects depending on the learner profile. Therefore, we had to create our own model to answer our RQ2.

3 MODEL FOR ANY EDUCATIONAL SYSTEM CONNECTED TO OUR JOINT SYSTEM

To answer our RQ2, we seek to treat 5 specific aspects:

- Activities granularities
- Playful and pedagogical aspects differentiation
- Genericity toward any educational system
- Simplicity of use (accessible to a non-expert)
- Connectivity to the joint system

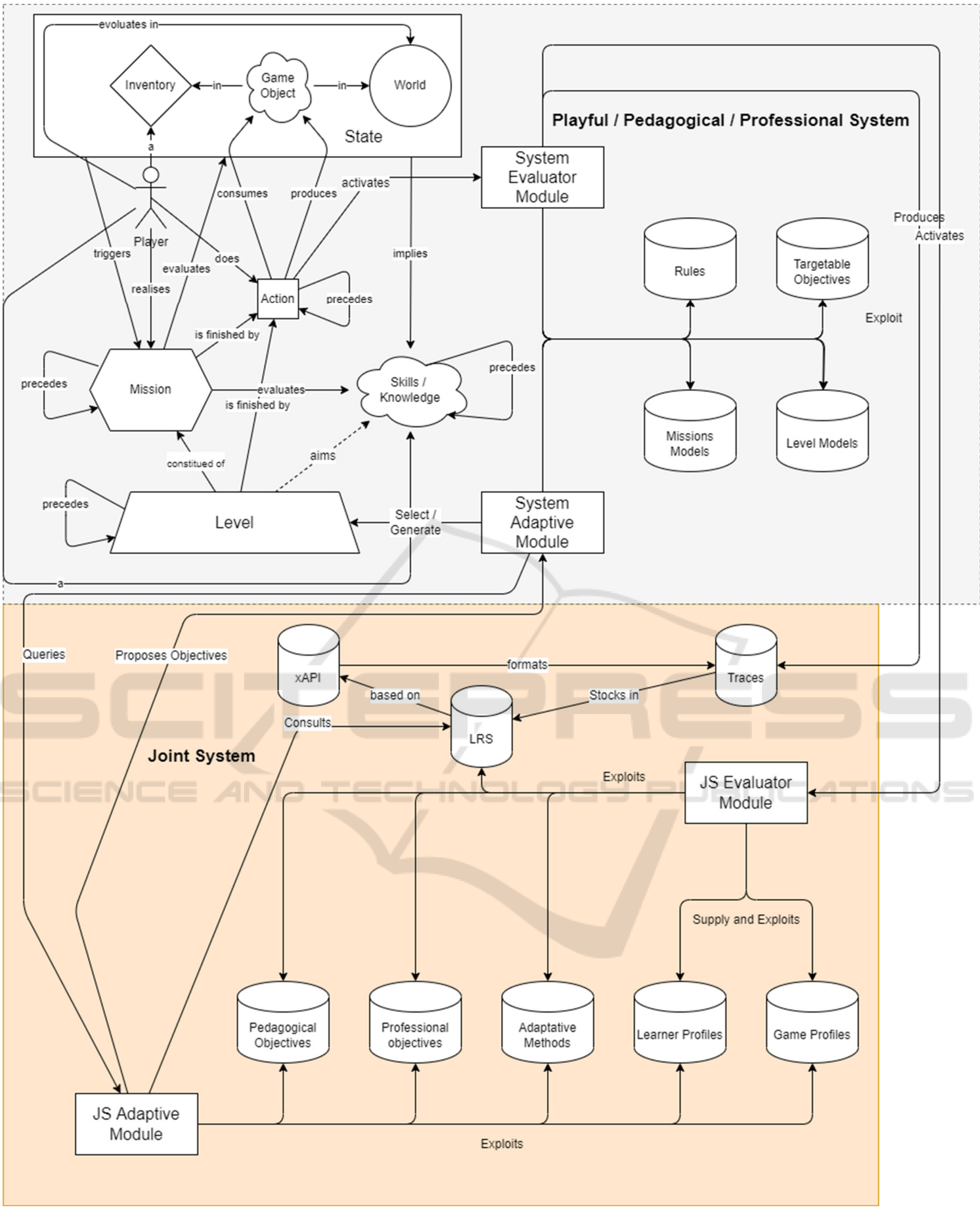

Figure 1: Model for an educational system and its connexion to the joint system.

To construct our model, we decided to take inspiration from both MPLOG (Guinebert et al., 2017) and MoPPLiQ (Marne et al., 2013). It is interesting to note that MoPPLiQ has been conceived along with “the 6 facets” which we evoked in section 2.1.

To manage the granularities of our activities we decided on a three degrees scale:

- Levels
- Mission
- Action

3.1 Levels

A Level can precede another Level and is composed of several Missions. There are three key elements to understand how to manage our Level degree. First, targeted skills and knowledge are primarily linked to Missions, not Levels. Therefore, since Missions are the components of our Levels, their targeted skills are linked to the Level they are a part of.

However, and it is our second element, the way Missions are presented to the learner is entirely up to the educational system we are modeling. Thus, it is entirely possible to represent Levels as an orderly sequence of mandatory Missions or as a multiset of free access Missions with no obligation for the learners.

For example, you could use our Levels to represent a quiz for which each question would be considered a Mission or use them to represent an open world game where each Mission would be playful objectives and quest disseminated throughout the land.

Finally, the third element to note toward our Level degree is the way we consider precedencies between Levels. Our model allows you to freely define your precedencies. There are no restrictions, be it an absence of precedencies, fixed precedencies such as in MoPPLiq, dynamic precedencies as in MPLOG or dynamic precedencies linked to only skills or both skills and system components.

3.2 Missions

Our Mission degree is here to represent a set of objectives for the learner. You can use them to represent any kind of objectives, be it professional, pedagogical, or playful. For example, achievement, quests, questions, rapports, meeting objectives, financial incentives, etc. could all be considered as Missions depending on the modeled system.

To determine the fulfillment of a Mission, the system must check if its current state has reached the objectives part of the Mission. The way precedencies work for Missions in our model, is the same as in our Level degree.

Missions can target several knowledge and skills. It is the fulfilment or not of those Missions by the learner that should indicate the system whether a learner mastered, failed, and/or experienced the targeted skills.

As mentioned earlier, Missions can be linked to either pedagogical, professional, or playful objectives. Moreover, we expressed both the hypothesis that equivalencies could be made between pedagogical and professional skills, and that several flow curbs should be considered in our systems (H2 and H3). Therefore, we consider here that skills and knowledges can be of either of these three natures.

3.3 Action

Our Action degree corresponds to the most atomic degree of our model. The learner evolves in Levels to fulfill a Mission, but ultimately, there is only one thing that he/she ever does: Actions. In the same way Roles worked in MPLOG, in our model, Actions consume and produce System Objects and nothing else.

What is interesting are the consequences of those production and consumption of Objects. To fully understand their reach, we must explain both what can be a System Object and how it affects the system. A System Object can be anything useful to the modeling of the scenario or the system's functioning. It can be game resources, Boolean flags, files, credits, points, given answers, etc.

Those Objects can then be found in two categories of Inventories like what can be found in MPLOG. They can be found in Environment/World/System Inventories which represent the objects available to every learner connected to the system and/or available to the system itself. They can also be found in personal inventories. The objects in those inventories are only related to the learner those inventories are linked to.

3.4 Connection to the Joint System

Our joint system must fulfill three purposes toward any system connected to it:

- Trace tracking
- Profile Evaluation
- Adaptive Scenarios

Since Actions are the only things done by learners, they are what drives any kind of evaluation and any kind of trace tracking. When an Action is done, it can, or not, lead to an evaluation by the system. It is defined by the modeled system.

Therefore, we make a distinction between the System Evaluation Module and the Joint System Evaluation Module. The first one is ad hoc to the modelled system and can be mustered by Actions, end of Levels, Missions' fulfilments, etc. The second evaluates normalized logs (in our case our own xAPI template, but you could use your own for your own joint systems) sent to the joint system's LRS.

Thus, when using our joint system model, the evaluation sequence is as follows:

1. An Action is done by the learner
2. This Action leads ultimately to an ad-hoc evaluation by the system
3. The system evaluation module generates a normalized log of the evaluation and Action and sends it to the LRS
4. The Joint System evaluation module acknowledges the log
5. It evaluates the impact on the learners involved and modifies their profile consequently

The adaptive part is similar but a little bit more direct. Once again, we have two different modules. A System adaptative module and a Joint System adaptative module. Both those modules are activated only when the modeled system considers it necessary; that is to say, when the modeled system estimates that an adaptation of its content could be useful to the learner. This adaptation could happen at the start of the session to determine the best Level for the player, to select the best Mission during a Level or even to generate new Missions and Level.

The JS adaptive module is modular itself. It can be perceived as a toolbox usable by the modeled system's module. The role of the JS adaptive module is to provide the various feedback from the tools queried by the module from the modeled system. In our case, our current adaptive module has only one tool in its box. The role of this tool is to establish, for a given learner, a priority order for any skills or knowledge that has been passed on to it. Our tool currently bases itself on four key aspects: precedencies as in CbKst (Doignon, 1994), scoring, system type and knowledge maintenance. Therefore, the sequence of actions between our two modules using this tool is as follows:

1. The System module signals a need for adaptation
2. It establishes a list of knowledge and skills that could be targeted and sends it to the Joint System Module
3. The Joint System module analyzes the impact each skill's mastery would have on the learner and ranks it by order of priority
4. It sends back the ranking order to the System Module which uses it in an ad hoc way to adapt the learner's scenario.

Thanks to the division of both the evaluation and adaptation modules as two separate entities, it is possible to connect any kind of educational system to our joint system as long as its ad hoc modules make the interface with our normalized ones. It also makes the independent evolution of the modeled and joint systems feasible. This last point is quite important in a company context where there is a need for constant production of new training systems and improvement of old ones. The normalization of the data by and to the joint system makes it possible.

3.5 Application and Conclusions

We are in a professional context. Therefore, we will probably never use this model to represent "pure" video games without any serious components. However, we need it for 100% pedagogical and/or professional application. Therefore, we asked ourselves if our system was generic enough to represent any kind of system with various degrees of playful, pedagogical, and professional components. Moreover, we also wanted to test our hypothesis which claims that to fully exploit our joint systems, it is necessary for our educational systems to be conceived and implemented with our joint system in mind.

Table 1: Conclusion on Genericity of our model and possibility of Connection to our JS on various kinds of systems.

	Genericity	Connection to JS
Hypothetical	OK	OK
Implemented	OK	OK
Publicly Available	OK	Only Evaluation

3.5.1 Genericity

To test our genericity, we established a list of 10 systems that we modeled without any connection to the Joint System. Of those 10 systems, 4 are hypothetical systems that could be useful for our company, 3 are systems we internally developed and conceived using our method and the last 3 are systems broadly available to the public.

2 of the hypothetical systems, 2 of the internally developed ones and 2 of the public ones (namely Voracy Fish (G Interactive, 2012) and ClassCraft (Sanchez et al., 2015)) are or would be generally classified as serious games. The third hypothetical system is a gamified system, and the third internally developed system is a quiz system with very limited playful elements. The fourth hypothetical system tries to emulate a pedagogical system that would give information on whether a learner succeeded, participated and was a present to a non-computer-mediated training. Finally, the last system we wanted to model was a pure video game that has often been transformed or used toward more pedagogical solutions (Bos et al., 2014; Ekaputra et al. 2013): Minecraft.

For the most open world and/or generic systems like Minecraft or Classcraft we had to establish generic Missions. Otherwise, we were able to model every system thus leading us to consider our approach as sufficiently generic.

3.5.2 Thoughts about the Connection to the Joint System

Afterwards, we pondered on how we would connect these models to the joint system. The hypothetical and implemented systems have been designed with our JS in mind. Therefore, there is obviously no challenge whatsoever to model their connection to it. For the three others, it would be possible to create an interface that would normalize their produced logs and link them to the associated skills and knowledge. Thus, even if a system has not been conceived to be connected to our Joint System, the evaluation part can be maintained. This is important because it means we can use those data to improve the information we have on a learner profile.

However, the same cannot be said for the adaptation part. We can distinguish two cases. First-case scenario, the modeled system has no adaptation module whatsoever. It is therefore impossible for it to use any information produced by our joint system.

Second-case scenario, the modeled system has an adaptation module. If it is possible to interact with the module, an interface could hypothetically be made to normalize the data coming from and toward our joint system adaptation module. But this interface remains hypothetical because there is no assurance that the exchange sequence, that we established in section 3.4 for the adaptation modules, will be followed by the module of the modeled system. In the case where it is not possible to communicate with the modeled module, it is, of course, not possible for the modeled system to use the Joint System adaptation module.

Therefore, only a handful of systems seems to be able to infirm our hypothesis H1 and their existence remain hypothetical. Thus, we can consider our hypothesis H1 to be mostly verified.

4 DESIGN METHOD

4.1 First Abstraction Level

As specified in section 2.1, to answer our RQ1, we posed two hypotheses, the first one being that it is possible to have equivalencies between pedagogical and professional objectives in such a way that “real” and simulated results can be freely exploited by any systems using said objectives.

The second one, which we attempted to address in section 3.5.2 is that it is necessary to take those equivalences and the connection to our Joint System into consideration in the design phases of any educational system destined to be connected to the Joint System.

None of the approaches we could find in the literature seemed to take both those hypotheses into consideration. As indicated in section 3.5, the gamified systems and learning games are not the only systems that could benefit from being connected to the joint system. Yet, most of the systems that are interesting to us comprise either playful and/or pedagogical/professional components.

Given those last two facts, there is an obvious interest for us to inspire ourselves from learning games and gamified design methods. However, it is also important to note that our method cannot be a simple fusion of said methods since we must consider both the joint system and the genericity of the method.

To conceive our method, we decided on an approach dividing it into several phases. Meaningful examples of such a division for methods can be found in the literature, notably in (Garcia et al., 2017), (Morschheuser et al., 2017) and (Avila-Pesantez et al., 2019). Thus, the first question we had to ask ourselves to design our method was: “Which division do we have to adopt in the case of a joint system’s design method?”

This division and its workflow are not trivial since our method aims to be able to help design any system including playful and/or educational elements that would be connected to our joint system. Indeed, if it is true that, the approach available in the literature, share common phases, they also have their differences toward their composition and/or articulation. Therefore, it is possible to take inspiration from them, but, as indicated in section 2.1, they cannot be used in their current state to answer our problematic.

To establish our phases, we listed the steps one needed to/could take in order to construct those systems. Those steps, that we will further detail in section 4.2, and the existing phases in the literature lead us to establish six different phases:

1. Preparation and Analysis
2. Context Determination
3. Junction and Constraints
4. Ideation and Design
5. Development
6. Evaluation

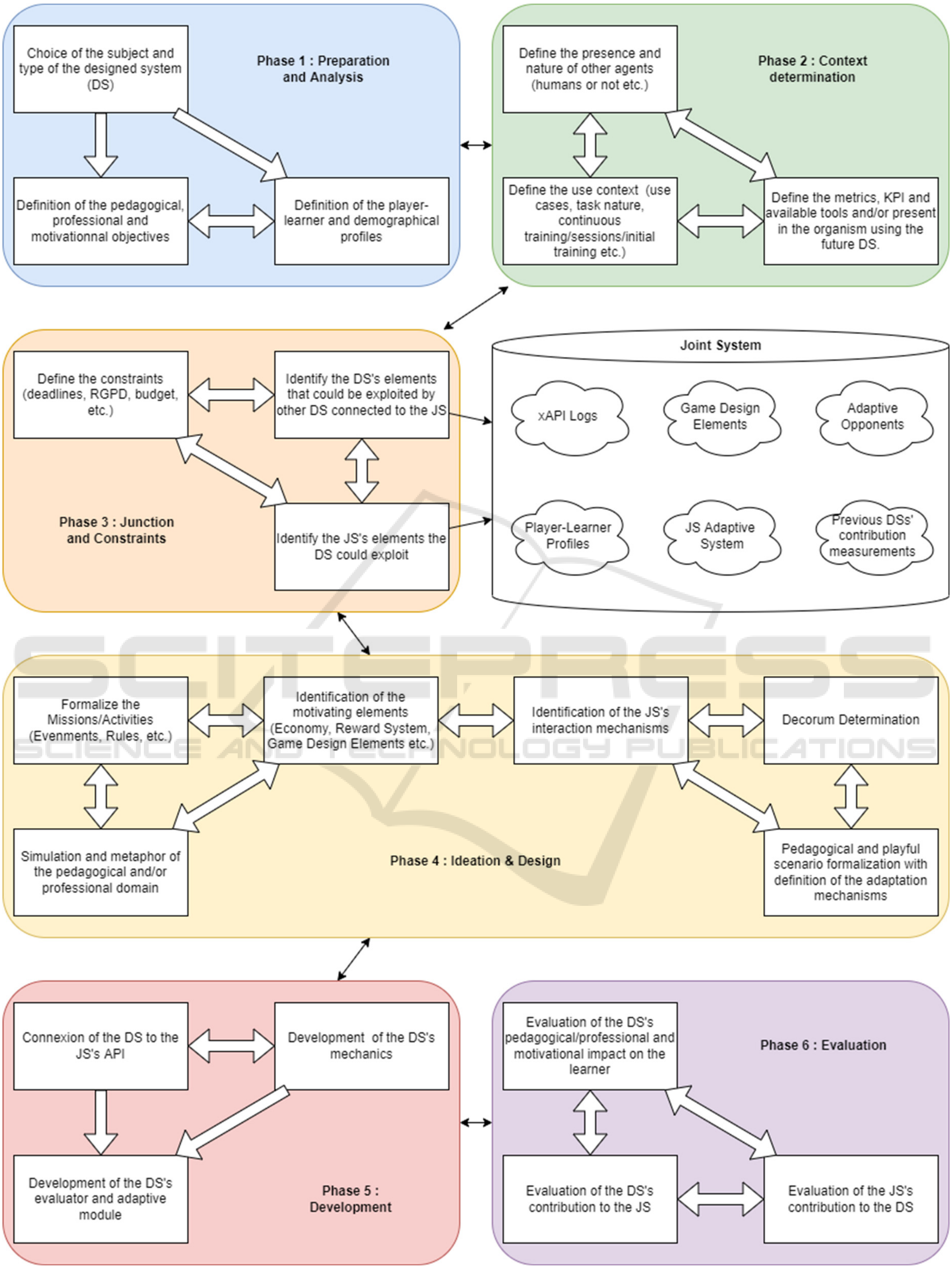

Figure 2: Method for the design and implementation of an educational system destined to be connected to a joint system.

In our case, the number of employees working on the design of those systems is little. Thus, we had to take this into consideration in the workflow of the phases. We decided to inspire ourselves from the learning games approach and make it quite free. As you can see it in Figure 2, it is possible to get back to previous phases from any of the more advanced phases. This allows a more agile approach for which multiple iterations are easier to handle. However, even with this “free” workflow, we consider it as a sequence: you must begin with the first phase and muster your way up to the sixth one.

4.2 Second Abstraction Level

We will detail what is to be done for each phase. We will also indicate our proposed workflow for its various steps.

4.2.1 Phase 1: Preparation and Analysis

This phase is common to almost every method or approach we could find. In this phase, one must define three key elements:

- The system’s type
- The system’s objectives
- The concerned learners’ profiles

Here, the system’s type refers to the kind of system you wish to design: a learning game, a gamified system, a quiz, etc. The very nature of this system is both what will determine the nature of the objectives (pedagogical, playful and/or professional) and the profiles that will be addressed by it.

Yet, it could be argued that it is the objectives that lead to the creation of the system, and therefore that the type of system is determined by the objectives one wishes to address. A similar reasoning can be found for the learner’s profile. This explains why we established a free workflow between those steps.

Those steps are the fundamental reason behind the design and the implementation of your system. This explains why we grouped them together in this phase.

For example, our juridical team wanted to teach some juridic aspects to our collaborators, thus, defining the targeted profiles as well as rhe pedagogical and professional objectives. We decided that a learning game would be the best way to provide the said teaching, therefore defining the system’s type. Finally, by choosing this type, we had to define playful objectives for our learning game.

4.2.2 Phase 2: Context Determination

A similar phase can be found in “the 6 facets” (Marne et al., 2012): in this phase the user must define the global usage context of his/her system. We distinguished three steps for which we established a free workflow:

- Define the presence and nature of other agents
- Define the context itself
- Define the metrics, KPI and tools available

The first step allows you to establish whether you intend your learners to interact with other learners or not and whether you want them to confront themselves to computer-controlled opponents. We consider it as part of the context because it directly impacts the way you can use your system (which is addressed by the second step).

In the second step, you must establish your system’s use case, the tasks’ nature, whether your system is for initial training only or not, whether you want to involve groups of synchronous plays or sequences of asynchronous ones, etc. In short, you must define the situations your system will be used in.

The third and final step of this phase is more related to the context of the organization itself. You must ask yourselves which tools, KPI and metrics are available to you and/or the learners and how you could use them jointly with your system.

The juridic game we cited as an example in 4.2.1 is a mono-player game with an asynchronous leaderboard and no computer-controlled opponent. It is not limited to the initial training of employees and can be linked to various home tools.

4.2.3 Phase 3: Junction and Constraints

This third phase has once again a free workflow and is composed of three steps:

- Define the constraints
- Identify the element exploitable by the joint system
- Identify the element exploitable by the designed system

Similarly to what can be found in (Morschheuser et al., 2017), the constraints refer to any kind of constraints one could apply toward the implementation of the system. What are your deadlines? What is your budget? Who is available to design and implement the system? What about GDPR/legal questions? etc.

The two other steps are only found in our approach. As of yet we only tested our method and joint system with the sharing and adaptation of skills and knowledge. However, we also think that game design elements and logs could be used to generate adaptive opponents and/or interfaces and are currently working on their addition to the JS adaptive module (see section 3.4).

In those two steps the designer must consider every information that will be produced by the system and all information that is available in the joint one to establish how the designed system could improve the others or be improved by them.

The juridic learning game that serves as our example needed to be developed quickly and with no additional funds besides salaries. It provided the JS with traces regarding the mastering or not of various juridic and playful skills and knowledge. By linking it to the JS it was able to use the learning adaptive module. This module implies the respect of GRPD laws, but also allows the learning game to exploit the other systems' traces for its adaptation.

4.2.4 Phase 4: Ideation & Design

This phase is composed of six different steps:

- Formalization of the Missions/Activities
- Simulation of the pedagogical/professional domain
- Identification of the motivating elements
- Identification of the interaction mechanics with the system
- Determination of the Decorum
- Pedagogical/Professional and Playful scenario formalization and determination of its adaptation mechanisms.

Formalizing the Missions and/or Activities (by using our model) allows you to establish the event and rules that will drive the system. You can directly link them to your objectives and use these formalizations to ensure that every profile and goal has been taken into consideration.

The simulation of the domain (that you can find in "the 6 facets") means that you must establish how your computer-mediated solution represents and simulate the activities and/or pedagogical-professional tasks.

Typically, the motivating elements that you must identify in the third steps are game design elements. Is there a reward system? How is it designed? What about the global economy of your system?

The interaction with the system (also found in "the 6 facets") establish and formalize the Actions a learner can make with the system. Those interactions codify what can and cannot be done by the user. It can directly be linked both to the formalization of the Missions and to the simulation of the domain.

The Decorum is also defined in "the 6 facets". It is mostly a playful component and is linked to the motivating elements. The decorum is defined by the graphical elements and narration of your system.

The final step helps you consider how the junction's steps of the third phase interact with every other step of the phase 4; notably, the motivating elements and Missions.

The juridic learning game was designed by inspiring ourselves from the goose board game. The Decorum was directly inspired by it and we divided the obtained board in steps according to the process we intended to simulate. The learboard served as a motivating element and the desire to beat it was modeled by a Mission. A session of the game is modelled by a Level and each part of the process is modelled by a Mission.

4.2.5 Phase 5: Development

We determined three steps that one needs to consider while developing the system:

- Connection to the joint system
- Development of the mechanics
- Development of the system's evaluation and adaptation module

The first and second steps can be done in any order you want. The first step stipulates that you must develop and consider the way your modules will connect to the joint system. Depending on the system's nature and/or programming language, you will be able to reuse previously implemented connection modules.

The second step is the development of the system itself. We won't delve into too many details for this step because we think it should be left to your decision which development method is the best.

Finally, the third and final step of this phase seek to implement the final step of phase 4 by relying on the implementations and development made in the first two steps of this phase.

4.2.6 Phase 6: Evaluation

Every method needs an evaluation phase. In our case we divided it in three steps with free workflow:

- Evaluation of the system's impact on the learner
- Evaluation of the system's contribution to the joint system
- Evaluation of the joint system's contribution to the designed system

Those three evaluations are closely linked together but do not rely on the exact same indicators. The system's impact can be measured either by looking into the profile's evolution or by looking at the learner's performance at similar tasks.

The designed system's contribution is directly linked to elements and information it shared with the joint system. Until those elements and information are used by another system connected to the joint system, the design system's contribution will remain poor. Yet those data can still be used to make reports on a learner's performance.

Similarly, if no previously connected systems shared information and elements useful to the designed system, the usefulness of the joint system will be kept at a low point for the designed system. However, since the joint system produces adaptive data useful to the designed system it keeps a modicum of usefulness even in this case.

The juridic learning game is currently the only system currently providing juridic traces to our JS. The evaluation of its impact on other systems is therefore limited at the moment. We were able collect data and evaluate the juridic skills of our collaborators. Moreover, the game was able to make full use of the adaptive module.

5 CONCLUSIONS

We seek to create a complex system that would allow data, game elements, learners' profiles, logs, etc. to be freely shared between systems of various natures that are used in various contexts. We also want this system to provide a set of normalized tools usable by pedagogical, playful and professional systems to adapt their content to the learner/player/user.

The creation of said system led us to ask ourselves two different Research Questions:

- RQ1: "Which approach is needed to guarantee both the efficiency and the relevance of a complex system composed of several serious games and gamified systems?"
- RQ2: "Which formalism or model to adopt in order to ease the adaptation and differentiation of playful and pedagogical scenarios?"

To answer our RQ2, we conceived a generic model that we used in three case scenarios. Firstly, we used it to model hypothetical systems that would be more or less playful. Secondly, we used it to model existing systems available in the literature or commerce. Thirdly, we used it to model and implement systems designed for our company.

Thanks to those three case scenarios, we concluded that our model seems to fulfill most of the five aspects we considered in order to answer our RQ2:

- Activities granularities
- Playful and pedagogical aspects differentiation
- Genericity toward any educational system
- Simplicity of use (accessible to a non-expert)
- Connectivity to the joint system

Of those five aspects, only the simplicity of use could not be tested yet. The results we obtained by testing the connectivity to the joint system seems to confirm H1 that stipulates that a system should be designed with the JS in mind to best exploit it. Indeed, our results seem to show that it is possible to exploit data from a system not designed to be used jointly with our JS, but that it would be difficult for this system to use the available adaptive tools.

As seen in section 2.1, the best way to answer our RQ1 seems to be the use of a dedicated method. Given our hypotheses and that this is still an ongoing research in the field, we decided to create our own method.

This method used jointly with our model allowed us the design and the implementation of three different systems. Those three still lack skill overlap with each other thus leading, at the moment, to their limited evaluation. However, the first results regarding the usability of the method and model are quite encouraging. Our current joint system adaptive module comprises a unique tool similar to a really lightweight ITS that provides feedback on demand to the connected system. To do so, this tool takes into consideration the skill nature, the memorization process, the precedencies between objectives and the leaner's global evaluation.

It is important to note that, as shown previously, since systems must be designed while knowing the various existing tools they can use, the addition of new tools to our adaptive module should mean that those shouldn't be usable by previously designed system.

If such a case were to happen, the only way for previously designed systems to use those new tools would be to reiterate our design method on phase 3.

Our future works will be driven by two different axes. Firstly, verify the equivalence between pedagogical and professional skills in our systems (H2). Secondly, improve the JS adaptive module

For our first axis, we aim to design and implement new gamified systems and training tools that could be used in parallel to our existing learning games. Those new systems would provide us with both information that could infirm or confirm H2. Moreover, with more systems implemented, we would, of course, have more knowledge and skills overlap and give more usefulness to our JS.

For our second axis, the improvement of the module is twofold. First, we need to upgrade our existing tool. To do so we intend to rely more on previously acquired data in order to change the way priorities are decided for our objectives. At the moment our current adaptive tool tries to improve knowledge maintenance by making the player/learner redo the activities at increasing intervals. We intend to further fine-tune this aspect in the future.

Secondly, we need to make our module more versatile for it to accept more adaptive tasks than just the objectives adaptation (note that an objective can be professional and/or playful and is not limited to pedagogical objectives). A way to reach this goal is to create new tools that could be used by our designed systems. For example, we intend to add adapted opponent generations by using the shared logs and profiles to establish adapted behavior and difficulties.

REFERENCES

Alsawaier, R. S. (2018). The effect of gamification on motivation and engagement. *The International Journal of Information and Learning Technology*.

Avila-Pesantez, D., Delgadillo, R., & Rivera, L. A. (2019). Proposal of a Conceptual Model for Serious Games Design: A Case Study in Children With Learning Disabilities. *IEEE Access, 7, 161017-161033*.

Bos, B., Wilder, L., Cook, M., & O'Donnell, R. (2014). Learning mathematics through Minecraft. *Teaching Children Mathematics, 21(1), 56-59*.

Bryan, J. (2006). Training and performance in small firms. International small business journal, 24(6), 635-660.

Caponetto, I., Earp, J., & Ott, M. (2014, October). Gamification and education: A literature review. In *European Conference on Games Based Learning (Vol. 1, p. 50).* Academic Conferences International Limited.

Csikszentmihalyi, M. (2000). Beyond boredom and anxiety. Jossey-Bass.

Dale, S. (2014). Gamification: Making work fun, or making fun of work?. *Business information review, 31(2), 82-90*.

Deterding, S., Björk, S. L., Nacke, L. E., Dixon, D., & Lawley, E. (2013). Designing gamification: creating gameful and playful experiences. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems (pp. 3263-3266).*

Doignon, J. P. (1994). Knowledge spaces and skill assignments. In *Contributions to mathematical psychology, psychometrics, and methodology (pp. 111-121)*. Springer, New York, NY.

Ekaputra, G., Lim, C., & Eng, K. I. (2013). Minecraft: A game as an education and scientific learning tool. *ISICO 2013*, 2013.

El Moudden, F., & Khaldi, M. (2018). Developing an ims-ld collaborative project creation application coproline (Collaborative Project Online). *International Journal of Engineering Applied Sciences and Technology, 3(4), 38-43.*

Garcia, F., Pedreira, O., Piattini, M., Cerdeira-Pena, A., & Penabad, M. (2017). A framework for gamification in software engineering. *Journal of Systems and Software, 132, 21-40.*

Guinebert, M., Yessad, A., Muratet, M., & Luengo, V. (2017, September). An ontology for describing scenarios of multi-players learning games: toward an automatic detection of group interactions. In *European Conference on Technology Enhanced Learning (pp. 410-415).* Springer, Cham.

Hummel, H., Manderveld, J., Tattersall, C., & Koper, R. (2004). Educational modelling language and learning design: new opportunities for instructional reusability and personalised learning. International *Journal of Learning Technology, 1(1), 111–126*

Interactive, G. E. N. I. O. U. S. (2012). Voracy Fish: New multiplayer serious game for physical rehabilitation (2012).

Kappen, D. L., & Nacke, L. E. (2013, October). The kaleidoscope of effective gamification: deconstructing gamification in business applications. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications (pp. 119-122).*

Landers, R. N. (2014). Developing a theory of gamified learning: Linking serious games and gamification of learning. *Simulation & gaming, 45(6), 752-768.*

Liu, M., McKelroy, E., Corliss, S. B., & Carrigan, J. (2017). Investigating the effect of an adaptive learning intervention on students' learning. *Educational technology research and development, 65(6), 1605-1625.*

Marfisi-Schottman, I. (2012). Méthodologie, modèles et outils pour la conception de Learning Games (*Doctoral dissertation*, INSA de Lyon).

Marne, B., Carron, T., Labat, J. M., & Marfisi-Schottman, I. (2013, July). MoPPLiq: a model for pedagogical adaptation of serious game scenarios. In *2013 IEEE 13th International Conference on Advanced Learning Technologies (pp. 291-293).* IEEE.

Marne, B., Wisdom, J., Huynh-Kim-Bang, B., & Labat, J. M. (2012, September). The six facets of serious game design: a methodology enhanced by our design pattern library. In *European conference on technology enhanced learning (pp. 208-221).* Springer, Berlin, Heidelberg.

Morschheuser, B., Hamari, J., Werder, K., & Abe, J. (2017). How to gamify? A method for designing gamification. In *Proceedings of the 50th Hawaii International Conference on System Sciences 2017.* University of Hawai'i at Manoa.

Nacke, L. E., & Deterding, C. S. (2017). The maturing of gamification research. *Computers in Human Behaviour, 450-454.*

Ouadoud, M., & Chkouri, M. Y. (2018, October). Generate a meta-model content for disciplinary information space of learning management system compatible with IMS-LD. In *Proceedings of the 3rd International Conference on Smart City Applications (pp. 1-8).*

Peng, H., Ma, S., & Spector, J. M. (2019). Personalized adaptive learning: an emerging pedagogical approach enabled by a smart learning environment. *Smart Learning Environments, 6(1), 1-14.*

Roussel, P., & Laboratoire interdisciplinaire de recherche sur les ressources humaines et l'emploi (Toulouse). (2000). *La motivation au travail: concept et théories.* LIRHE, Université des sciences sociales de Toulouse.

Sanchez, E., Young, S., & Jouneau-Sion, C. (2015, June). Classcraft: de la gamification à la ludicisation. In *7ème Conférence sur les Environnements Informatiques pour l'Apprentissage Humain (EIAH 2015) (pp. 360-371).*

Villiot-Leclercq, E. (2007). La méthode des Pléiades: un formalisme pour favoriser la transférabilité et l'instrumentation des scénarios pédagogiques. *Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, 14(1), 117-154.*