

Design and Development of an e-Learning Project Management System

Modelling and Prototyping

<https://doi.org/10.3991/ijet.v15i19.14595>

Maha Khaldi (✉), Mohamed Erradi
Abdelmalek Essaâdi University, Tetouan, Morocco
khaldi.94.maha@gmail.com

Abstract—In this work, based on the results obtained concerning the analysis and the needs of our project which concerns the design and development of an e-Learning project management system, we present the modeling stage with UML. Based on one hand, on two diagrams: class diagram and use case diagram, for static modeling and on the other hand, based on three diagrams: activity diagram, sequence diagram, diagram transition state for dynamic modeling. Finally, we offer examples of models for our project.

Keywords—E-Learning project management system, modeling with UML, mock-ups.

1 Context of the Work

In an IT project, it is imperative to base yourself on a structured approach that describes its progress. The choice of project management turns out to be a decisive phase in the realization of the project in good conditions. It is necessary to define a work methodology, as well as a development process and to deduce a timetable for the project to follow. Indeed, to be lucid, we analyzed several research works which worked on the e-learning, thus, we retained especially research which relates to the sharing and the adaptation of educational documents in e-learning using technology of the Semantic Web, the design and / or development of systems, whether resource production or intelligent architecture in the educational field [1; 2; 3].

In this work, based on the results obtained regarding the analysis and the needs of the project, we present the modelling stage with UML of our project which concerns the design and development of a project management system e- Learning. Indeed, concerning the analysis carried out around our project on the one hand, we were interested in the choice of the methodology to follow to carry out our project. Our choice of methodology was based on the ADDIE method (Analysis Design Development Implementation Evaluation), the method most used in e-learning projects regardless of their scale [4]. According to the ADDIE model, the conduct of an e-Learning project requires five phases [5]. On the other hand, this analysis

allowed us to have a clear idea on the products that exist in the market, on the different functionalities which they contain, and of the way with which we must approach the design and the development of our system [6].

Concerning the needs of our project after having defined our choices in development process and modelling language of our project, we have defined, the different stages of an e-Learning project which constitute the stages of the project, the different actions for each stage, the participation of the different actors in each stage, the different roles in the different actors of the project by citing for each actor the different tasks to be carried out for each stage of the project, the different actors with each stage and each action (status) and the participation of the different actors involved in each action of an e-Learning project [7].

The different definitions of the realization are presented in the form of tables treating the various actions for the realization of the project, allowed us to have a clear idea to pass to the stage of the conception concerning the planning of the tasks, the proposal of the different diagrams necessary for the realization of the project and the proposal of models for the different interfaces. The following table summarizes the results obtained concerning the participation of the different actors involved in each action of an e-Learning project [8].

Table 1. Intervening in each action of an e-Learning project

Actors	Scoping note	Graphic charter	Educational design	Prototype	Synopsis	Storyboard	E-learning module
Administrator							
Project manager	✓	✓	✓	✓	✓	✓	✓
Content Expert						✓	
Educational engineer		✓		✓	✓	✓	
Art Director		✓	✓				
Graphic designer		✓			✓		
I Integrator						✓	
Multimedia specialist						✓	
Speaker				✓			✓
Translator				✓			✓
Computer scientist	✓			✓			✓
Customer	✓	✓	✓	✓	✓	✓	✓

2 Development Process

The development process is a determining factor in the success of a project because it frames its different phases and characterizes the main features of its conduct. For this, the choice of a development method, which is adequate to the specifics and requirements of a project, must be developed beforehand in order to obtain a quality project that meets the needs and expectations of users. There are several unified processes like Extreme Programming (XP) and Rational Unified Process (RUP). In the case of this project, we chose to work with the 2TUP process [9] (2 Track Unified Process, also called Y development model) because it targets projects of any size and

it has been able to make a large place in the technology and project risks. It provides a response to the constraints of continual evolution of information systems by breaking down the analysis of the system along a functional and a technical axis [10].

3 Modelling Language

Modelling is the design of a model. According to its objective and the means used. In computer science, we talk about data modelling to designate a stage in the construction of an information system [11; 12]. The 2TUP process relies on UML (Unified Modelling Language) throughout the development cycle because the different diagrams of the latter allow for their ease and clarity, to model the system well at each stage. In fact, UML is defined as a graphic and textual modelling language intended to understand and describe needs, specify, design solutions and communicate points of view [13].

UML unifies both object-oriented notations and concepts. It is not a simple notation, but the concepts transmitted by a diagram have a precise semantics and carry meaning in the same way as the words of a language, this is why UML is sometimes presented as a method when it is absolutely not. UML also unifies the notations necessary for the different activities of a development process and offers, through this, the means of establishing the monitoring of the decisions taken, from the definition of needs to coding [14; 15]. It results from the unification of proven techniques for the analysis and design of large software and complex systems.

From this fact, we can conclude that the UML language helps us at all stages of a project, as it thus offers us many advantages for the analysis and the design of a system. Thus, The UML couple and the unified process offer an approach to drive the realization of object-oriented systems [16; 17; 18].

Likewise, the UML language provides several diagrams to allow the modelling of a system from specification to implementation. UML represents a software system according to two complementary aspects: static or structural modelling and dynamic or behavioral modelling.

- Static modelling is based on the use case diagram, the class diagram, the object diagram, the component diagram, and the deployment diagram.
- Dynamic modelling is based on the state diagram, the activity diagram, the collaboration diagram and the sequence diagram.

4 Modelling with UML

4.1 Static system view

In the case of this project, two diagrams (class diagram and use case diagram) to be considered for static modelling:

Class diagram: A class diagram generally expresses the static structure of a system, in terms of classes and the relationships between these classes. This is the

most common diagram found in object-oriented modelling. Generally, it is used to model, system vocabulary, dependencies and database schemas. Class diagrams are important for viewing, specifying and documenting structural models. The class diagram of our project is presented in Figure 1.

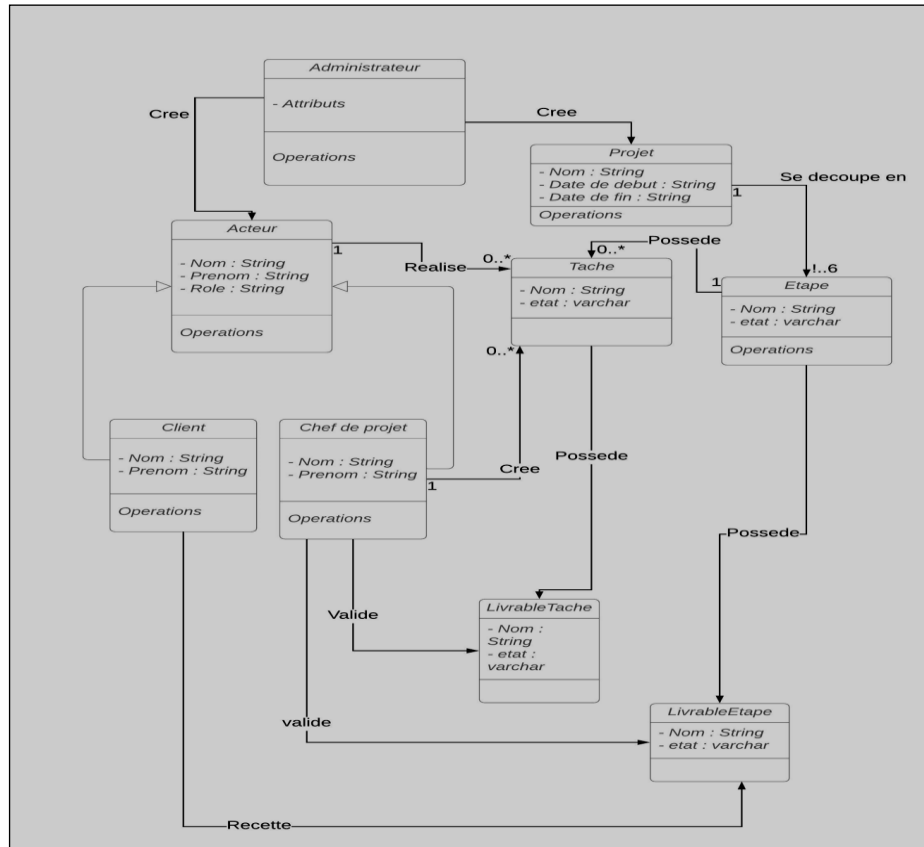


Fig. 1. Class diagram

The class diagram represents the different models of structure which visualizes the attributes and the main operations to realize the structure. In this project, we have specified nine classes (Administrator, Project, Actor, Task, Stage, Customer, Project Manager, Deliverable Task, Deliverable) and each class contains the possible operations and the necessary attributes.

- The Administrator class presents two main operations, manage actor and manage project. This gives the administrator the possibility to create and modify actors and projects.
- The Project class contains the following attributes: Name; Start date; End date, where each project is divided into 6 stages.

- The Actor class has as attributes: name; first name; role, where each actor can perform tasks.
- The task class has as attributes: name; state, where each task belongs to a stage.
- The Stage class has as attributes the different stages which are six in number.
- The Customer class is made up of the attributes: surname and first name and which in turn validates the deliverables.
- The Project Manager class has as attributes: name; first name, and as operations: affects; create; modify the tasks and validate deliverables.
- The Deliverable Task class has as attributes: name; the state of the deliverable and as operations: add; modify the deliverables. These deliverables are validated by the project manager.
- The Deliverable Stage class has as attributes: name; the state of the deliverable and as operations: add; modify the deliverables. These deliverables are validated by the project manager.

Use case diagram: Use case diagrams (workflows) describe, in the form of actions and reactions, the functionality of the system as well as its behavior from a user's point of view. They make it possible to define the limits of the system and the relations between the system and the environment. A use case is a specific way of using a system. It is the image of a functionality of the system, triggered in response to stimulation from an external actor. The use case diagram for our project is presented in Figure 2.

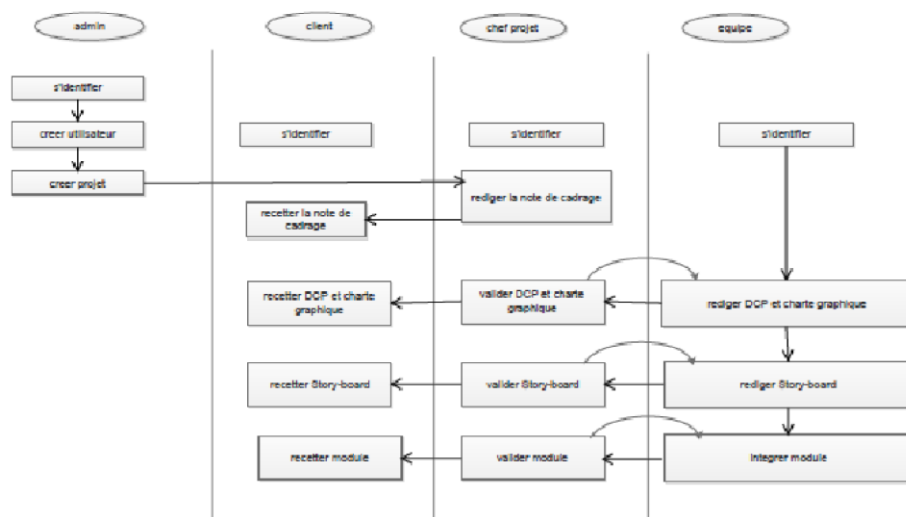


Fig. 2. Use case diagram (workflow)

To create the case diagram, i.e. the project workflow, we have specified all the steps that an e-learning project can have. Indeed, there are four responsible parties in

this flow which is made up of an administrator, a client, a project manager and a team where:

- The administrator can create projects, and assign them to a project manager.
- The project manager in turn writes the scoping note for each project and validates the other deliverables (storyboard, synopsis, graphic charter, educational design document).
- The client receives each deliverable after it has been validated by the project manager.
- A team that writes the deliverables and then integrates the different parts of the e-learning module after validation by the project manager and the client.

4.2 Dynamic view of the system

In the case of this project, three diagrams (activity diagram, sequence diagram, transition state diagram) to be considered for dynamic modelling:

Activity diagram: The activity diagrams allow you to focus on the treatments. They are therefore particularly suitable for modelling the flow of control flows and data flows. They thus make it possible to graphically represent the behaviour of a method or the course of a use case. Figure 3 shows the activity diagram of the digital learning module.

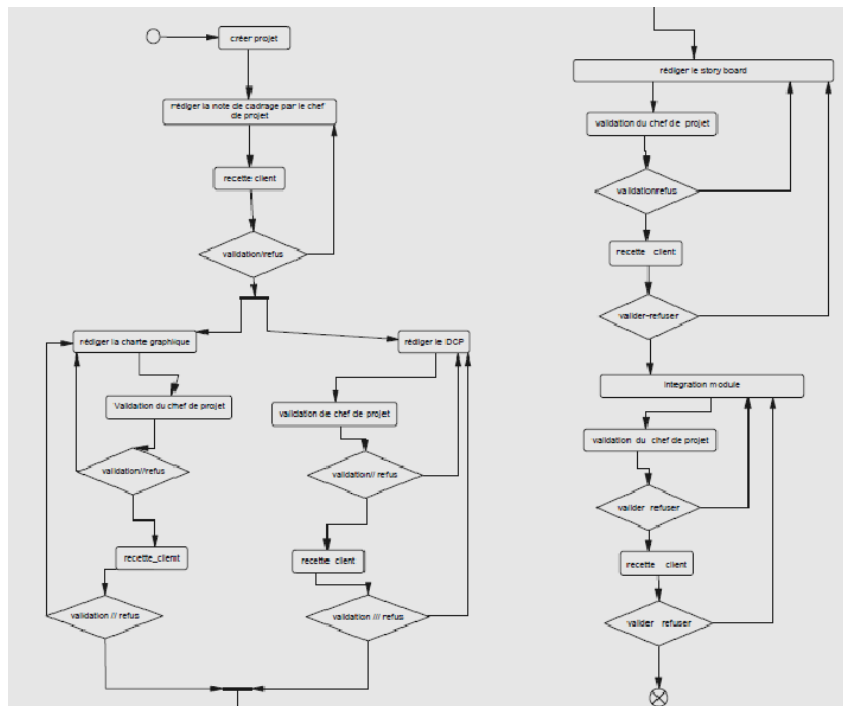


Fig. 3. Activity diagram of the digital learning module

Figure 3 with its two parts expresses the different stages through which an e-learning module goes, which begins with the creation of a project and ends with the integration of the module and its validation. After the creation of the project, the project manager produces the scoping note. If the latter is validated, we move on to the parallel drafting of the educational design document and the graphic charter, which must be validated by the project manager and the client otherwise, we must re-edit the non-validated ones. After we go to the writing of the storyboard which is in turn must be validated by the project manager and then by the client otherwise, we must modify it. Once validated, we move on to integration.

Sequence diagram: A sequence diagram represents the objects participating in a particular interaction and the messages they exchange, organized in time sequences. Concerning the sequence diagram, for organizational and management questions, we propose in this work only the summary sequence diagram of the e-learning project which includes all the sequence diagrams of each phase (framing, educational design, prototype, writing modules, production / integration, deployment / test). Figure 4 presents the summary diagram of the e-learning project.

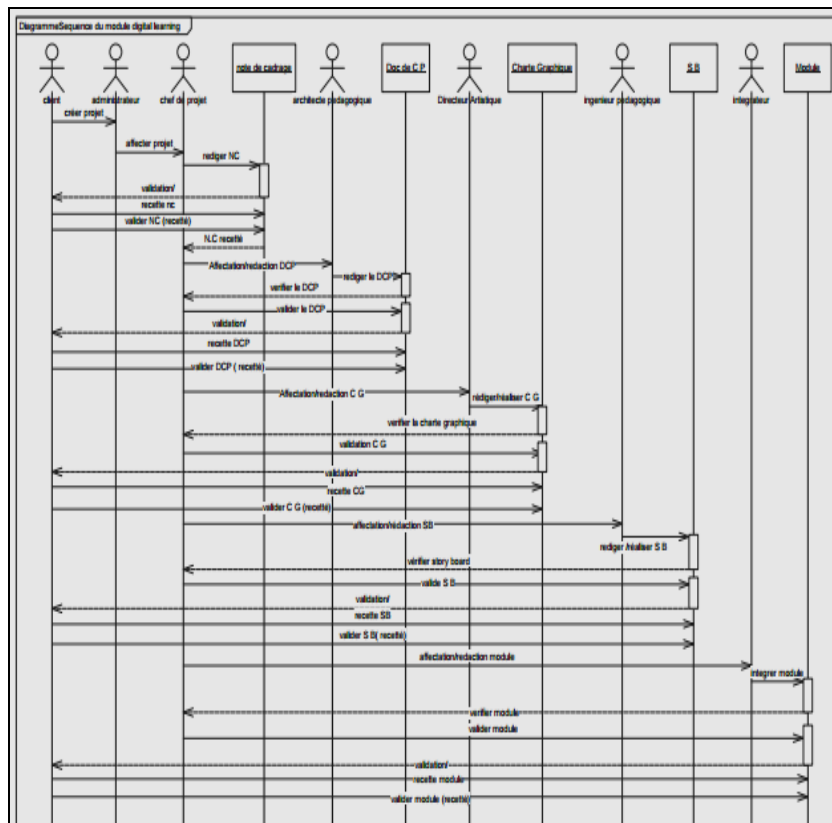


Fig. 4. Sequence diagram: digital learning module

Figure 4 which represent the sequence diagram of the e-learning project, allows visualizing the dialogue between the different actors. It makes it possible to represent the objects making up the e-learning project; in this case the objects are the deliverables. Indeed, each deliverable must be validated by the project manager on the one hand and by the client on the other. Thus, the educational design document is produced by the educational architect, the graphic charter is produced by the artistic director, the storyboard is produced by the educational engineer, then the integration of the project is carried out by the integrator.

Diagram of transition states: Transition state diagrams are used to describe the behavior of a system. They make it possible to describe the changes of state of an object or a component, in response to interactions with other objects / components or with actors.

Concerning the transition state diagram, as in the case of sequence diagram, We only present the transition state diagram of the digital learning module, the diagrams for each document (scoping note, educational design, graphic charter, synopsis, storyboard, integration, graphic design) will be presented in the appendix. Figure 5 shows the transition state diagram for the e-learning project.

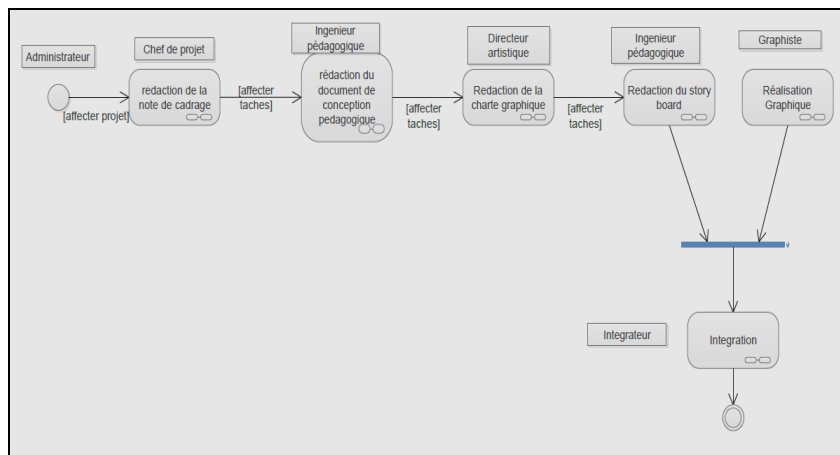


Fig. 5. Diagram of transition states of the e-learning project

The transition state diagram of the e-learning project represents the different sequences of an e-learning project as well as the possible actions on each sequence, where the sequences are summarized in the writing of each deliverable.

5 Interface Models

5.1 Mock-up

In man-machine interface and computer ergonomics, paper mock-up (or prototyping) is a technique still widely used in user-centered design. This technique

helps project stakeholders (designers, developers, salespeople, etc.) to design software or a website that meets the expectations and needs of end users [19]. They are most often the first raw ideas thrown on paper, made of sketches and drawings, representing the arrangement and the zones of contents (or zoning). The zoning is the basis for the work of the graphic designer. The latter covers the different areas of the screen thus defined according to the visual identity of the product. It specifies the role and position of each zone and defines their relative size as well as their visual importance [19].

Likewise, we will speak of a functional model or wireframe which concerns the diagram used when designing a user interface to define the zones and components that it must contain. The wireframe consists concretely of a sketch, a paper collage or a digital diagram [20]. For Jakob Nielsen, wireframe modelling corresponds to horizontal prototyping [21], which is to say to the development of the graphic part of the man-machine interface. It allows:

- Confirmation of satisfaction of the user interface requirements and the possibilities of the product to be produced (software, website).
- The simulation of the a priori functioning of the product to be produced according to the initial objectives.
- A first estimate of the development time and cost.

The wireframe serves as a discussion support with the sponsors of the product, but also for carrying out user tests. In this case, we will speak more readily of a prototype [22].

5.2 Examples of models

In this section, we offer examples of the proposed models. Axure Rp7 [23] is the technological tool used for the production of these models. Indeed:

Figure 6 presents the mock-up authentication interface to connect to the interface of each user. Or there are two fields: A field for the name and another field for the password, which allows him to " reach your workspace, in addition to a logo and a validation button.



Fig. 6. User interface model

In the following, we offer two examples of models. Figure 7 concerns the project management model. Figure 8 concerns the model of the management of actors who belong to the administrator interface.

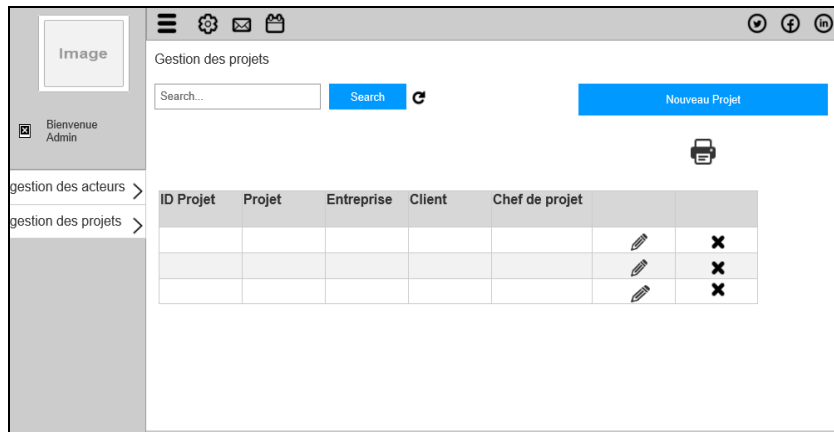


Fig. 7. Project management

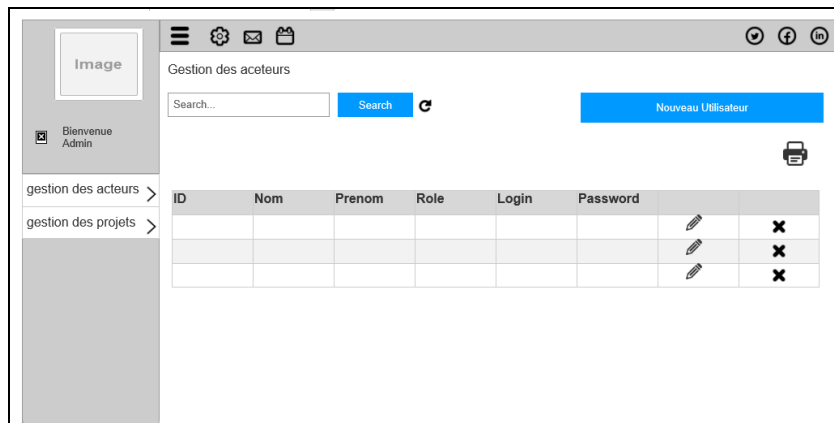


Fig. 8. Management of actors

6 Conclusion

As a conclusion to this article, after having mentioned the UML modelling of the project, we have presented the different diagrams required for the realization of the project. At the level of static modelling, we presented the class diagram which presents all the classes of the project and the interactions which exist between the different classes. We have presented the use case diagram for the project workflow, detailing all the steps that an e-learning project can have. There are four responsible

parties in this flow which consists of an administrator, a client, a project manager and a team. We have clarified the roles of each responsible party in the project.

In terms of dynamic modelling, we presented the activity diagram which expresses the different stages through which an e-learning project goes, which begins with the creation and ends with the integration of the project and its validation. We presented the sequence diagram of the e-learning project, which allows visualizing the dialogue between the different actors. It allows you to represent the objects making up the e-learning project. We have presented the transition state diagram of the e-learning project which represents the different sequences of an e-learning project as well as the possible actions on each sequence.

In terms of layout, we have presented some models which relate to certain project interfaces which represent the different interfaces and the main functionalities of each role.

7 References

- [1] Chekry Abderrahman, et Khaldi Mohamed, (2013). Semantic Web Technologies for the Reuse and Adaptation of Educational Documents in E-Learning, *International Journal of Engineering Pedagogy (iJEP)*, 2013, Volume 3, Issue 1, 52-55. <https://doi.org/10.3991/ijep.v3i1.2167>
- [2] LFAQIHI Brahim, DAOUDI, Najima, AJHOUN Rachida, (2018). Design of an Intelligent Educational Resource Production System, *International Journal of Emerging Technologies in Learning (iJET)*, Vol 13, No 12, <https://doi.org/10.3991/ijet.v13i12.8914>
- [3] Najoua Hrich, Mohamed Lazaar, and Mohamed Khaldi. (2019). MAPSS an Intelligent Architecture for the Pedagogical Support: Experiment, Results & Discussion. *International Journal of Emerging Technologies in Learning (iJET)*. Volume: 14, Numéro : 21, Pages, 61-79. <https://doi.org/10.3991/ijet.v14i21.11012>
- [4] Basque, J. (2004). How does ICT change the teaching engineering practices of the university professor? "Educational engineering in the age of ICT: practices and research", *International Journal of Technologies in University Teaching*, (Profetic), 1 (3). 7-13.
- [5] E-learning.prestataire.com/conseils/projet-e-learning - the steps.
- [6] Maha Khaldi, Mohamed Erradi (2018), Design and development of an e-Learning project management system: Analysis of the situation, *International Journal of Scientific & Engineering Research* Volume 9, Issue 8, pp129-131.
- [7] Maha Khaldi and Mohammed Erradi. 2019. Design of an e-Learning project management system: Project needs. In *Proceedings of the 4th International Conference on Big Data and Internet of Things (BDIoT'19)*. Association for Computing Machinery, New York, NY, USA, Article 40, 1–5. <https://doi.org/10.1145/3372938.3372978>
- [8] <https://www.techno-science.net/definition/670.html> (02/23/2020).
- [9] P. Roques and F. Vallée, *UML 2 in action*. 2nd Edition, EYROLLES. N ° 11213, 2002, 400 pages.
- [10] Olivier Guibert (2007), *Course on Analysis and Design of Information Systems (Tools and Models for Software Engineering)*, Computer Science Department of the IUT of the University of Bordeaux 1, France.
- [11] Dutreuil, S., Huneman, P., Varenne, F. & Silberstein, M. (2014). Introduction. Modeling and simulation: arguments and lively questions. In: Franck Varenne éd., *Modéliser &*

- simuler - Tome 2: Epistemologies and practices of modeling and simulation (pp. 1-18). Paris: Material Editions. <https://doi.org/10.3917/edmat.varen.2014.01.0002>
- [12] Benoit Charroux, Aomar Osmani et Yann Thierry-Mieg, UML2 pratique de la modélisation, Synthex - Synthèse de cours et exercices corrigés 2010.
- [13] P. ROQUES. – UML 2: modéliser une application web. N 12389, 4e édition, EYROLLES, 2008, 246 pages.
- [14] P. ROQUES. – Mémento UML 2.5., EYROLLES, N° 14356, 2015, 14 pages.
- [15] <http://www.uml-sysml.org/uml/>.
- [16] Object Management Group (OMG). Unified modelling language: Superstructure. Internet, août 2004 <https://www.omg.org/spec/UML/>. <https://doi.org/10.3403/bsisoiec19505>
- [17] Object Management Group (OMG). Object constraint language. Internet, May 2006. <http://www.uml.org/>
- [18] https://fr.Wikipedia.Org/wiki/Maquettage_papier?
- [19] Nogier, J-F. (2008), Software ergonomics and web design: The user interface manual, 4th edition, Dunod.
- [20] [https://fr.Wikipedia.org/wiki/Wireframe_\(design\)](https://fr.Wikipedia.org/wiki/Wireframe_(design)).
- [21] Nielsen, (1993), Usability Engineering (ISBN 0-12-518406-9).
- [22] Jonathan Arnowitz, Michael Arent, Nevin Berger (2007) Effective Prototyping for software makers. <https://doi.org/10.1016/b978-012088568-8/50003-5>
- [23] <https://www.azure.com>.

8 Authors

Maha Khaldi is a PhD student of Computer Science and University Pedagogical Engineering, She is currently with S2IPU, Abdelmalek Essaadi University, Tetouan, Normal School Superior, Tetouan, Morocco (e-mail: khalidi.94.maha@gmail.com).

Mohamed Erradi was born in Morocco. he is a professor of computer science and University Pedagogical Engineering at the Normal School Superior, Tetouan, Morocco. he is an active member on S2IPU, Abdelmalek Essaadi University, Tetouan. (e-mail: erradi.mo.mo@gmail.com).

Article submitted 2020-05-30. Resubmitted 2020-06-24. Final acceptance 2020-06-26. Final version published as submitted by the authors.