Original Article

Use of the agile scrum method to manage an e-learning platform in the face of the application of the ADDIE model in the pedagogical engineering of a training course

Lamya Anoir[1], Mohamed Khaldi[2], Mohamed Erradi[3]*

*Research team in Computer Science and University Pedagogical Engineering Higher Normal School, Abdelmalek Essaadi University, Tetouan – Morocco. (*Corresponding author)*

E-mail:[1]lamya.anoir@uae.ac.ma; [2]medkhaldi@yahoo.fr; [3]erradi.mo.mo@gmail.com

Abstract

With the advent of new technologies and the changing needs of learners, however, it has become essential to explore more flexible and agile approaches to the design and management of e-Learning courses. This article examines how the agile Scrum method can be used to manage an e-Learning platform, and how it can be integrated with the ADDIE model to improve the pedagogical engineering of a training course. The two methods, Scrum and ADDIE, act on the two axes of e-Learning platform management: IT project management and pedagogical content engineering. Consequently, the aim of our work is to highlight the different issues that are taken into account when working with the agile Scrum method and the ADDIE pedagogical model, and to present an example of the application of agile methodologies in the pedagogical design of personalized, adaptive learning systems. With its emphasis on flexibility and rapid iteration, this hybrid approach aims to respond effectively to the changing needs of learners and technological developments. In this article, we present the challenges of each method and propose our approach to integrating these methods into the technical and pedagogical management of a learning platform.

Keywords: Agile Scrum, ADDIE, integration, e-Learning, pedagogical engineering.

1. Introduction

With the development of Information and Communication Technologies (ICT) continuing to accelerate, the growing and ever more demanding needs of users and a constantly changing economic context, IT projects are becoming increasingly complex. As a result, managing e-Learning projects is becoming a delicate issue for any company, and their mastery and success are essential, whatever the size or type of project.

Digital transformation implies the inclusion and development of agile methodologies that make it possible to adapt and personalize the way of working to project conditions, achieving flexibility to the specific circumstances of the environment by working in a personalized and collaborative way (Lamya *et al*., 2021).

On the one hand, and in practice, the application of a process and management will accompany the designer through the various phases of project management, from the definition of objectives to the realization of deliverables, through resource allocation and schedule management. Among the most widely used methods are agile methods, which respond to traditional methods that are too predictive and inflexible, by introducing new, more flexible principles such as anticipation, self-regulation, feedback and collaboration. Scrum is the best-known of the agile methods. Created in 1996 by Ken Schwaber, it emphasizes the close-knit aspect of a self-organized team working towards a shared goal. Scrum's distinctive feature is that it places the end-user at the heart of the team, valuing the individual, the team, concreteness, application, collaboration and adaptation. Scrum is not an acronym but the English word for scrum in a rugby match, It's not a method in the strict sense of the word but rather an approach, a process framework and a set of principles, almost a philosophy based on change, result-oriented culture, transparency and communication, respect for users, customers and team spirit (Collignon & Schöpfel, 2016).

On the other hand, when we talk about the design of an e-Learning project we often talk about pedagogical engineering, which corresponds to a process implemented to solve a training or learning problem carried out by one or more actors driven explicitly or implicitly on the basis of principles derived from different theories, during which different tools are used to create different intermediate productions and whose end result is a learning environment proposed as a solution to the training problem. Among the models often used is the ADDIE model, in which analysis, design, development, implementation and evaluation are the successive stages in the product development process. This approach follows a regular, step-by-step process. It is currently the most widely used approach. It enables novices to follow a production path with deliverables at each stage.

Both the Scrum and ADDIE methods act on the two axes of e-Learning management: IT project management and pedagogical content engineering. They are new, more agile modes of project management that foster perceived autonomy and competence by providing the necessary tools and opportunities for self-regulated learners to adjust their learning strategies.

In the next sections, we will present our theoretical framework in order to identify our objective, and then propose a scheme that implements project management using the Scrum method and the design of a training device using the ADDIE model.

2. Materials and Methods

Our article focuses on two essential points. The first concerns the management of an e-Learning project using the Scrum method, and the second concerns the pedagogical design of an e-learning device using the ADDIE model.

2.1. Overview of e-Learning project management :

An e-Learning project is a human activity that achieves a clear objective on a timescale. It's distance learning using ICT.

The "learner" - a literal translation of "learner", a more appropriate term in this case - has all the essential elements for learning at his or her workstation. So many tools for learning alone without being isolated. Creating an e-learning course is a long, complex and energy-intensive process, involving a wide variety of participants (content experts, instructional designers, web and multimedia developers, etc.) (Reiss, 2007).

This type of project poses a number of challenges: effective coordination between team members, designing activities and materials adapted to online learning, and implementing all this on the e-Learning platform that offers MOOCs (Massive online Open Courses). It can be used in a variety of ways, including as an alternative or complement to face-to-face training. An e-Learning platform corresponds to a multi-channel digital environment, including videoconferencing, assessment, interactive laboratory and forum functions. For the pedagogical content of this model, personalized pedagogical follow-up (Lamya et al., 2020) is easy to implement.

An agile method is an iterative, incremental approach, performed in a collaborative spirit, with just the right amount of formalism. It generates a high-quality product while taking into account changing needs (Messager, 2009).

Studies show that the Agile method improves software development quality and productivity, reduces time-to-market (Reifer, 2002; Li et al., 2010; Cardozo et al., 2010) and enhances customer satisfaction (Boehm & Turner, 2003).

The Agile method is characterized by leadership, creativity and motivation of developers, technical excellence and simplicity of design, close collaboration between software developers and business teams, face-to-face communication, delivery of different parts of the software development at short and regular intervals, acceptance of changes in customer requirements during all phases of the development process in order to better satisfy the customer (Misra et al., 2009; Dingsoyr et al., 2012).

An agile method is an iterative, incremental approach, conducted in a collaborative way with just the right amount of formalism. It generates a high-quality product while taking into account evolving customer needs.

Agile methods are iterative, incremental software development models that aim to best meet the needs expressed by requesters, offering a high degree of responsiveness to their requests during development (Manifesto, 2001).

There are three roles in Scrum:

- The Product Owner (PO): is the product owner. He provides a shared vision of the product and is responsible for defining and managing product priorities.
- The Scrum Master: responsible for helping the team apply Scrum to its context, and has a duty to eliminate obstacles that may slow down the team's work.
- The Scrum team: is responsible for product development, and organizes itself to optimize its productivity and increase its skills to achieve this.
  Scrum is based on two artefacts, the pivots of activities:

- The product backlog: this is a list of "users stories", i.e. functional chunks of value that can be developed in a sprint (Bergier, 2011). This list represents the functional scope of the product. It is not set in stone and evolves over time according to requested modifications and new priorities encountered.
- The sprint backlog: this represents the part of the product backlog under development in

the current sprint. It is the set of tasks to be carried out during the sprint.

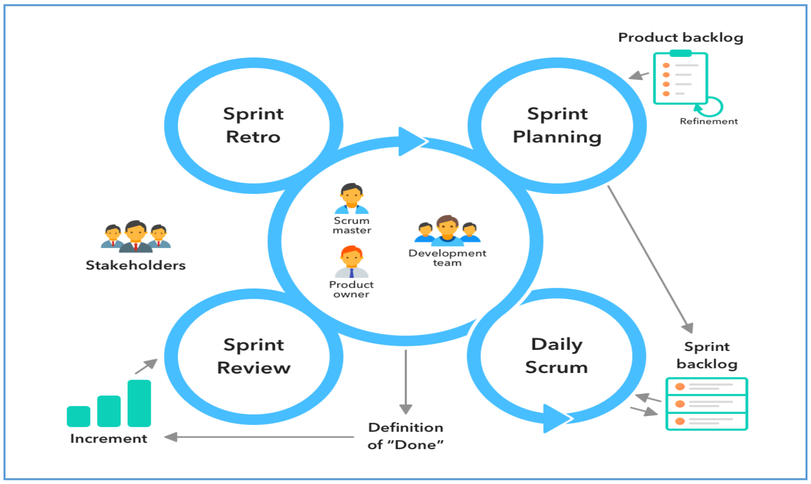

Figure 1: the SCRUM agile method

Figure 1 shows the steps involved in the Scrum process. The Scrum process is framed by five ceremonies (meetings defined within the Scrum framework):

- Release planning: a meeting to prioritize user stories and allocate them to the project's sprints.
- Sprint planning: a meeting designed to break down the user stories of the sprint to be launched into short development tasks.
- Scrum meeting: a daily 15-minute meeting at which each team member reports on what they did the previous day and what they're going to do today.
- The sprint review: a meeting aimed at showing the partial functional product. The PO can then compare the product with his request and, if necessary, propose modifications. Taking modifications into account means adjusting the product backlog and therefore the content of each sprint. In fact, the schedule may have to be adapted to take account of the modifications requested.
- The retrospective: a meeting that only concerns the project team and aims to identify what is working well and what needs to be changed, in order to collectively find solutions.

Scrum is an agile method dedicated to project management. The aim of this management method, or rather project management framework, is to improve team productivity. In an agile project, there's no need to specify and plan the entire target product. A first objective is set in the short term, and its realization begins immediately. When this first objective is reached, we stop, take stock, note possible improvements, set the next objective and start again in a realization phase... and so on until the final product is obtained. In concrete terms, with an agile project management method, the customer describes his vision of the product and the functionalities he would like to see included. This list of features is then submitted to the development team, which provides an estimate of the cost of implementing each of them.

Scrum is an agile project management methodology characterized by its flexibility and ability to adapt to changing project needs. It is based on a structured framework of roles, events and specific artefacts. In the context of managing an e-Learning platform, Scrum can be adapted to meet the needs of instructional engineering. Scrum roles such as Product Owner, Scrum Master and development team can be reassigned to suit the actors involved in creating an e-Learning platform.

2.2. Pedagogical engineering of an e-Learning project using the ADDIE model:

Many training programs claim to offer e-learning to their participants. The fact that it is accessible online does not in itself justify that the program has been built with learning in mind. The pedagogical strategy for delivering this content must be based on a well-thought-out development process. Various notions can guide the different educational strategies used.

Pedagogical engineering, formerly instructional design (Basque, 2017), also called instructional design (Trestini, 2016), learning design (Baron, 2011), "consists in studying, designing, realizing and adapting teaching devices, training, or courses" (Trestini, 2016). It defines all the elements of a course or training program, including its context, the knowledge and skills to be acquired, the scheduling of activities, the resources used and its delivery (Paquette, 2002). It therefore involves many actors linked to learning (instructional designers, learning facilitators, content experts, evaluators, learners, learning system applicants or even researchers), technologies (mediators) or project management (Basque, 2017). Pedagogical engineering attempts to respond to the complexity of device design (Chachkine, 2011). It follows phases defined by engineering models that may be circumscribed to a specific field (e.g. education) or be more general (Harvey and Loiselle, 2009).

The ADDIE model is the historical model for e-learning training engineering. More than 100 e-learning engineering models are based on ADDIE, with some variations and adjustments. ADDIE is the acronym of the five classic phases in the lifecycle of a learning system (Basque, 2017), which gave it its name from 1995: Analysis, Design, Development, Implementation and Evaluation (Clark, 1995).

Synthetically, analysis measures the difference between the current state and the desired state (Lepage *et al*., 2015) on different levels of the project, such as the training need, target clientele, context, constraints or learning resources (Basque, 2017). Design involves making choices among the various components of the device (objectives, strategies, media, pedagogy, techniques, evaluation methods, content organization and presentation). Development is the stage of realizing the device. Implementation (or dissemination) is the process of making the system available. Evaluation measures success and potential for improvement.

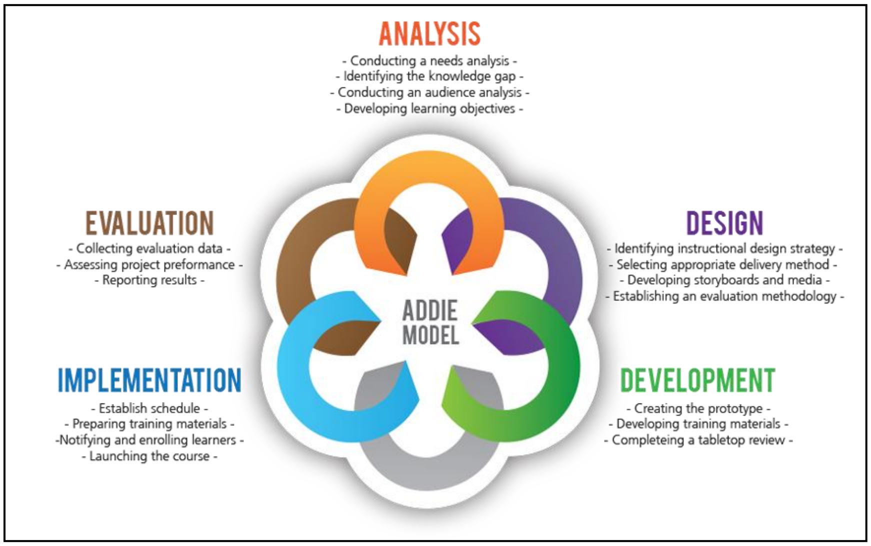
ANALYSIS
- Conducting a needs analysis -
- Identifying the knowledge gap -
- Conducting an audience analysis -
- Developing learning objectives -
EVALUATION
- Collecting evaluation data -
- Assessing project preformance -
- Reporting results -
DESIGN
- Identifying instructional design strategy -
- Selecting appropriate delivery method -
- Developing storyboards and media -
- Establishing an evaluation methodology -
ADDIE
MODEL
IMPLEMENTATION
- Establish schedule -
- Preparing training materials -
-Notifying and enrolling learners -
- Launching the course -
DEVELOPMENT
- Creating the prototype -
- Developing training materials -
- Completeing a tabletop review -

Figure 2: ADDIE model (Analysis, Design, Development, Implementation and Evaluation)

The design and management of an e-learning program is based on analytical or pragmatic methods/approaches. The pedagogical, technical, economic and organizational specifications are an essential phase to be developed from one of these approaches. Storyboarding is an important step in defining the various deliverables. The ADDIE approach is the easiest to implement for less experienced people. It defines the different phases of the project.

The waterfall model, like the ADDIE model, is used in many companies to produce e-learning modules. However, this way of working poses a number of problems for instructional designers. These include extended development cycles, communication with business experts and stakeholders, lack of time for the testing phase. To overcome these difficulties, we decided to integrate Scrum, which uses a sprint method for incremental development. A key feature of Scrum is its emphasis on organizing cross-functional teams and defining each role and responsibility in the development process, combining elements of the ADDIE model and implementing them in two-week sprints.

3 Results and Discussion

3.1 The parallels between management methodologies and the pedagogical design of an e-Learning project:

The ADDIE and Scrum frameworks are two development methodologies used to guide teams in the implementation of e-Learning projects. The philosophies of ADDIE and Scrum share many of the same practices. The Agile Scrum and ADDIE methodologies include analysis, design, development, implementation and evaluation as part of their process, but the Agile methodology has distinct characteristics that set it apart from ADDIE. Incremental organization, flexible planning, collaborative and transparent processes are the hallmarks of a project using Agile methodology.

Scrum is the best-known agile framework, used by companies of all sizes to manage ongoing projects and workflows. Where ADDIE requires each design stage to be completed before moving on to the next, Scrum is iterative, allowing different stages to be worked on simultaneously. If a department is swamped with e-learning project requests, Scrum can help prioritize tasks to respond more quickly to demand. It emphasizes collaboration, transparency and deliverables, making it an excellent continuous workflow model for tackling both large and small projects. The emphasis on accountability means that managers can easily see what everyone is working on, and maximize efficiency.

3.2 Adopting the SCRUM approach to designing an e-Learning project according to the ADDIE model

The application of some of the best practices learned from the hugely successful Agile/Scrum software development to the ADDIE model for instructional systems design.

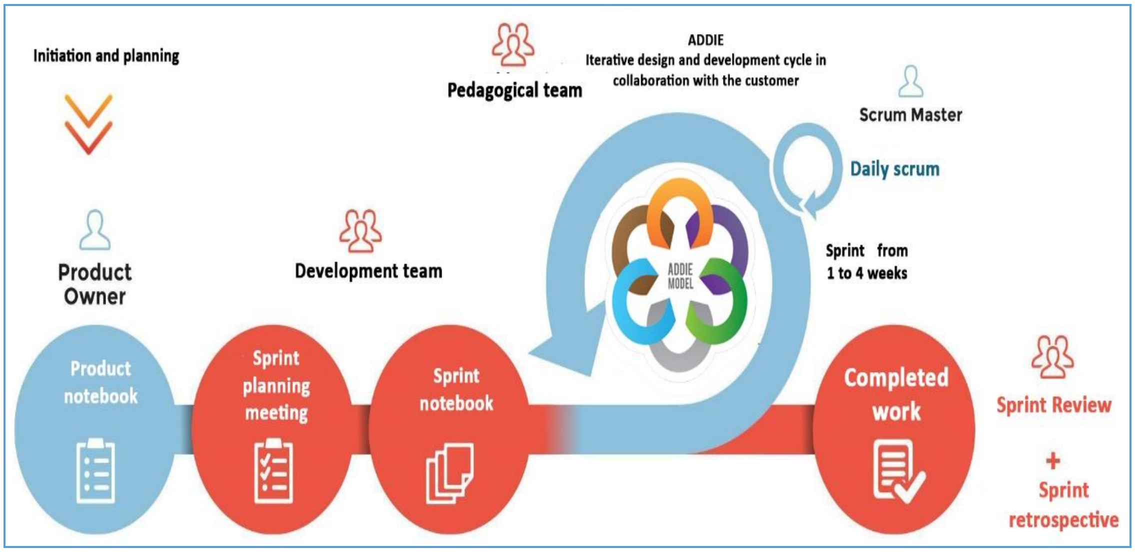

Figure 3: ADDIE and Scrum for designing and managing e-Learning projects

Figure 3 suggests a hybrid project management approach where certain phases at the beginning and end of a project are linear, and project execution (design and development) is iterative in nature.

This framework comprises five phases: initiation and planning, sprint planning meeting and implementation of a sprint log, iterative design and development according to the ADDIE model, deployment of work, sprint review and retrospective.

Initiation and planning: The first phase is linear and includes project definition, identification of business and educational needs, specific project goals and objectives, stakeholder identification, high-level assumptions and constraints, high-level budget and schedule, risks, dependencies and acceptance criteria. This phase includes high-level planning.

Sprint planning meeting: one of the most important stages of a Scrum project, it consists exactly in going through its objectives and its place in the project and its organization, which will also serve to answer two important questions:

- What can be done in this sprint?
- How can the selected product backlog items be transformed into a completed increment?

The maximum duration of sprint planning is 8 hours for a 4-week sprint. This time is proportionately less if the sprint duration is shorter. All sprint planning is grouped together in a notebook called the Sprint Notebook.

Iterative design and development according to the ADDIE model: Project execution takes place during the third phase, which is iterative by nature.

The product owner (Product Owner PO), in collaboration with business stakeholders and the educational team, maintains a hierarchical product backlog containing refined user stories. During release and sprint planning meetings, the project team selects the amount of work it thinks it can achieve in each sprint, and the number of sprints needed to properly deliver or release the feature.

Each feature generally goes through iterative ADDIE cycles (Analysis, Design, Development, Implementation and Evaluation). In this stage, the features published in the various release cycles are integrated to form an end-to-end solution. The product is deployed on the test system running the Learning Management System (LMS) to ensure quality and meet user needs, while individual functionalities are demonstrated and tested at the end of each release.

Deployment of work: Once the solution has been properly tested and end-user feedback incorporated, the solution is ready to be deployed on the production system and go live. As every project brings changes, this phase usually includes communication and change management to ensure wider adoption. The end of this phase marks the closure of the project and the transition to operations.

Retrospective sprint review: this phase may include an evaluation survey against the project's success criteria and a formal acceptance signature from the sponsors.

This example illustrates the process of integrating the ADDIE model and the Scrum method into the pedagogical engineering of a training course:

- **Pedagogical Analysis (ADDIE) and Scrum Planning:**

The team begins with an in-depth analysis of learners' needs, identifying existing gaps in educational resources. The Product Owner, in collaboration with the pedagogical experts, defines the learning objectives. These are integrated into the Scrum product backlog, with priorities determined according to needs.

- **Iterative Design and Development (ADDIE and Scrum):**

Initial lesson design is based on ADDIE Model Analysis. Using Scrum, two-week iterations are planned. The development team creates a first version of the lesson, gathers feedback from pedagogical experts and learners, then makes adjustments in the next iteration. This iterative process continues until the lesson achieves the desired quality.

- **Continuous evaluation (ADDIE and Scrum):**

After the initial deployment of a section of the course, evaluations are collected via surveys and learner feedback. The Scrum Master organizes regular retrospective meetings to discuss the feedback with the team. Based on this information, changes are made to future iterations, enabling continuous improvement in content quality.

- **Scrum Meetings and Stakeholder Communication**

The team organize regular Scrum meetings to discuss progress, obstacles and necessary adjustments. The Scrum Master communicates this information to stakeholders at dedicated meetings. For example, if changes in learners' needs are identified, the Product Owner adjusts the product backlog accordingly, ensuring immediate responsiveness.

- **Reduced Development Time (Scrum):**

Using Scrum, a team succeeded in reducing the overall development time of an online course from 30 weeks to 24 weeks. Rapid iterations enabled problems to be detected and resolved quickly, avoiding major schedule delays.

These examples demonstrate how the integration of ADDIE and Scrum enables more agile, responsive course design focused on the real needs of learners, while respecting the pedagogical objectives defined in the Analysis phase of the ADDIE model.

3. The actors in the ADDIE and SCRUM integration cycle:

In the integration of the ADDIE model and the Scrum agile method for the pedagogical engineering of a training course, several players play key roles in ensuring the success of the project. Here are the main players and their roles in this process (Table 1):

Table 1: Actors in the ADDIE and SCRUM integration cycle

Actors	**Roles**	
	SCRUM	**ADDIE**
Product Owner	The Product Owner is responsible for the project vision, defining learning objectives and creating the product backlog. He or she identifies learners' needs and prioritizes functionalities or content elements according to their importance.	The Product Owner plays an essential role in the Analysis phase of the ADDIE model, identifying learning objectives, learner needs and establishing the project vision.
Scrum Master	The Scrum Master is responsible for ensuring that the Scrum team follows Scrum principles and processes. He facilitates meetings, removes obstacles and supports the development team.	The Scrum Master contributes to coordination and communication within the project team, fostering greater collaboration when applying the ADDIE model.
Development Team	The development team is made up of instructional designers, developers, graphic designers and other experts involved in creating learning platforms. They are responsible for implementing the tasks identified in the product backlog.	The development team implements the phases of the ADDIE model, such as design, development, implementation and evaluation, using Scrum iterations to continually adjust and improve content.
After integration		
Learner	Learners play an essential role by providing feedback, comments and evaluations throughout the process. Their needs and reactions	

	are taken into account to adjust and improve the pedagogical content.
Pedagogical Experts	Pedagogical experts, such as pedagogues, educational psychologists and training specialists, provide advice on instructional design, the development of learning objectives, and the evaluation of pedagogical effectiveness. They collaborate with the Product Owner and the development team to ensure that pedagogical methods are appropriate.
Stakeholders	Stakeholders, who may include the educational institution's management, project sponsors, and other stakeholders, support the project, provide resources and strategic guidance. They must be kept informed of project progress and decisions.

The integration of ADDIE and Scrum implies strong collaboration between these players to ensure that pedagogical objectives are met, that content is of high quality, and that the process is agile and responsive to the changing needs of learners. Each of these players plays a crucial role in the successful pedagogical engineering of a training course based on this hybrid approach.

The ADDIE model is often criticized for its linearity, which can make it difficult to adjust along the way in response to learner feedback or technological developments. Integrating Scrum with ADDIE alleviates this problem by introducing shorter iterations and opportunities for constant revision and improvement. For example, after the Analysis phase of the ADDIE model, Scrum can be used to rapidly iterate on Design, Development and Implementation according to the changing needs of the learner.

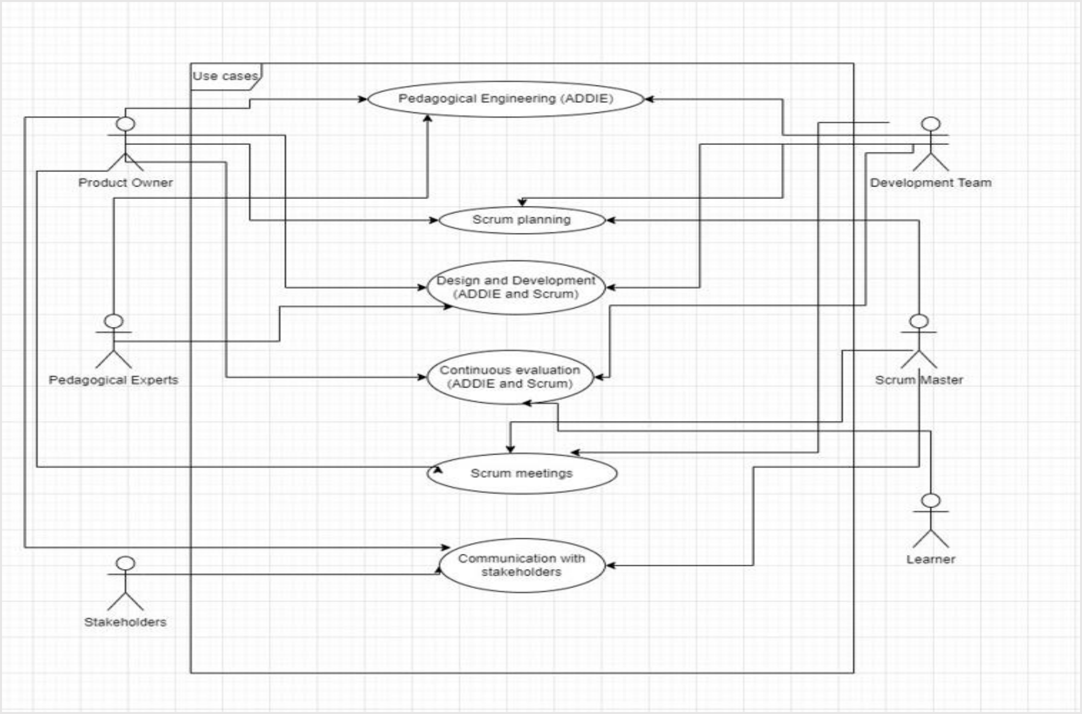
Use cases
Pedagogical Engineering (ADDIE)
Product Owner
Development Team
Scrum planning
Design and Development (ADDIE and Scrum)
Pedagogical Experts
Scrum Master
Continuous evaluation (ADDIE and Scrum)
Scrum meetings
Learner
Communication with stakeholders
Stakeholders

Figure 4: A use case diagram to illustrate the process of integrating the ADDIE model and the Scrum method

Figure 4 present a use-case diagram to illustrate the process of integrating the ADDIE model and the Scrum method, which involves representing the actors, the use cases and the interactions between them.

Use cases:

1. Pedagogical engineering (ADDIE)

   - Actors: Product Owner, Pedagogical Experts, Development Team

   - Description: Identify learners' needs, define pedagogical objectives and establish the project vision.

2. Scrum planning

   - Actors: Product Owner, Scrum Master, Development Team

   - Description: Draw up the product backlog, plan iterations (Sprints), and define specific objectives for each iteration.

3. Design and Development (ADDIE and Scrum)

   - Actors: Product Owner, Development Team, Pedagogical Experts

   - Description: Perform design and development tasks in successive iterations, integrating feedback from learners and pedagogical experts.

4. Continuous evaluation (ADDIE and Scrum)

   - Actors: Product Owner, Development Team, Learners, Pedagogical Experts

   - Description: Collect regular feedback, evaluate progress, and adjust pedagogical content according to the results obtained.

5. Scrum meetings

   - Actors parties: Scrum Master, Product Owner, Development Team

   - Description: Organize regular Scrum meetings to discuss progress, obstacles, and plan next iterations.

6. Communication with Stakeholders

   - Actors: Product Owner, Scrum Master, Stakeholders

   - Description: Maintain constant communication with stakeholders to share updates, gather feedback and ensure support.

   - The "Scrum Planning" use case is linked to the "Pedagogical Analysis" and "Design and Development" use cases to ensure planning aligned with pedagogical needs.

   - The "Design and Development" and "Continuous Evaluation" use cases are interconnected to enable rapid iterations and continuous content improvement.

This diagram provides a visual overview of the process of integrating the ADDIE model and the Scrum method into the pedagogical engineering of a training course. It highlights the interactions between the players and the stages in the process, making it easier to understand the hybrid approach.

4 Conclusion

The creation and management of e-Learning platforms is a long, complex and energy-consuming process, involving a wide variety of stakeholders (content experts, instructional designers, web and multimedia developers, etc.). This type of project generates a number of challenges: effective coordination between team members, design of activities and materials adapted to e-learning, implementation of all this on the platform used. The ADDIE methodology organizes practices into five sequential phases, with each phase the subject of a process designed to achieve a sufficiently solid definition to propel e-learning teams into each subsequent phase of the learning design process. They focus on quality at the end of the process, in the "Evaluation" phase. It is during this phase that e-learning teams and project stakeholders work together to determine the effectiveness of the work product and make adjustments. This contrasts with Agile methodology, which works in "sprints" or "iterations", short two-week periods during which the e-learning team implements and delivers an increment of work product for evaluation to ensure quality. With the Agile method, everyone is involved, including learners, designers, managers, customers and stakeholders. Effective collaboration is therefore crucial. Our proposal is a presentation of an approach that enables the IT management of an e-Learning project and the pedagogical engineering of the project with the two methodologies Scrum and ADDIE in particular when creating e-Learning solutions to improve performance. As well as providing continuous visibility and communication, managing an e-learning project with this framework also means that the practice of iterating as required helps to respond to and manage risks in a way that complements instructional design models.

Declaration of Conflict of interest

The authors declare that they have no conflict of interest.

Authors' contribution

All authors contribute equally.

References

Agile Manifesto, 2001. Disponible en avril 2012 à l'adresse http://agilemanifesto.org/

Baron, G. L. (2011). Learning design. Recherche et formation (68), 109-120. doi:10.4000/rechercheformation.1565

Basque, J. (2017). TED6312 – Ingénierie pédagogique et technologies éducatives [PDF]. Repéré àhttps://ted6313v2.teluq.ca/teluqDownload.php?file=2014/04/19_TED6313_Texte_Intro_IP.pdf

Begier, B. Quality assessment of an expert system: an instrument of regular feedback from users. In Nguyen N. T. Transactions on computational collective intelligence III, Springer-Verlag, 2011, 199-214.

Boehm, B., & Turner, R. (2003). Balancing Agility and Discipline: A Guide for the Perplexed, Portable Documents. Addison-Wesley Professional.

Cardozo, E., Neto, J. B. F. A., Barza, A., França, A., da Silva, F. (2010). SCRUM and productivity in software projects: a systematic literature review. In14th International Conference on Evaluation and Assessment in Software Engineering (EASE).

Chachkine, E. (2011). Quels scénarios pédagogiques pour un dispositif d'apprentissage à distance socioconstructiviste et de conception énonciativiste en FLE? (Thèse de doctorat, Université de Provence-Aix-Marseille I). Repéré à https://tel.archives-ouvertes.fr/tel-00604871/

Clark, D. (1995, juillet). History of the ADDIE Model [HTML] (dernière mise à jour le 5 septembre 2015). Repéré à http://www.nwlink.com/~donclark/history_isd/addie.html

Collignon, A., & Schöpfel, J. (2016). Méthodologie de gestion agile d'un projet. Scrum–les principes de base. I2D Information, donnees documents, 53(2), 12-15.

Dingsøyr, T., Nerur, S., Balijepally, V., Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, 85(6), 1213-1221.

Harvey, S. et Loiselle, J. (2009). Proposition d'un modèle de recherche développement. Recherches qualitatives, 28(2), 95-117.

Lamya, A., Kawtar, Z., Mohamed, E., & Mohamed, K. (2020). Personalization of an educational scenario of a learning activity according to the learning styles model David Kolb. Global Journal of Engineering and Technology Advances, 5(3), 099-108.

Lamya, A., Mohamed, K., & Mohamed, E. (2021, March). Personalization between Pedagogy and Adaptive Hypermedia System. In International Conference on Big Data and Internet of Things. Cham: Springer International Publishing.

Lepage, M., Sauvé, L., Plante, P. et Renaud, L. (2015). Analyse de besoins sur l'utilisation des outils Web 2.0: éléments essentiels pour la communication dans les équipes de recherche universitaire. Nouveaux c@ hiers de la recherche en éducation, 18(1), 87-113. doi:10.7202/1033731ar

Li, J., Moe, N. B., & Dybå, T. (2010). Transition from a plan-driven process to scrum: a longitudinal case study on software quality. In Proceedings of the 2010 ACM-IEEE international symposium on empirical software engineering and measurement (p. 13).

Messager, V., 2009. Gestion de projet: vers les méthodes agiles, 2nd ed, Architecte logiciel. Editions Eyrolles, Paris.

Misra, S. C., Kumar, V., & Kumar, U. (2009). Identifying some important success factors in adopting agile software development practices. Journal of Systems and Software, 82(11), 1869-1890.

Paquette, G. (2002). L'Ingénierie pédagogique : pour construire l'apprentissage en réseau. Québec, QC : Presses de l'Université du Québec.

Reifer, D. J. (2002). How good are agile methods? Software, IEEE, 19(4), 16-18.

Reiss, G. (2007). Project Management Demystified (3 ed.). London, UK: Taylor & Francis Ltd.

Trestini, M. (2016). Théorie des systèmes complexes appliquée à la modélisation d'environnements numériques d'apprentissage de nouvelle génération (Thèse de doctorat, Université de Strasbourg). Repéré à https://hal.archives-ouvertes.fr/tel-01432687/