Conceptualizing an Inductive Learning Situation in Online Learning Enabled by Software Engineering

Ouariach Soufiane[1], Khaldi Maha[2], Khaldi Mohamed[3]
A Research Team in Computer Science and University Educational Engineering-
Higher Normal School of Tetouan, Abdelmalek Essaadi University, Morocco[1, 3]
Rabat Business School-Rabat International University, Rabat, Morocco[2]

***Abstract*—Our work highlights the importance of adopting a systematic and methodical software engineering approach to the development of information technology projects for e-learning. We place particular emphasis on conceptualizing pedagogical scenarios and an inductive online learning situation. To ensure effective management of the information systems development process, we applied instructional design principles and adopted the 2TUP process, a refined version of the Rational Unified Process (RUP) suitable for projects of all sizes. To provide a visual representation of the system architecture and inform instructional design decisions, we used the Unified Modeling Language (UML) to create class, use case, activity, and sequence diagrams. We aim to demonstrate the potential of a structured software engineering approach to creating effective and efficient e-learning systems by conceptualizing an inductive online learning situation and five concrete examples illustrating the system's functionality. Our work underlines the importance of using standardized modeling languages such as UML to facilitate communication between stakeholders and collaboration between instructional designers and software developers.**

Keywords—Software engineering approach; instructional design; conceptualization scenario; online learning situation; inductive approach

I. Introduction

In an IT project, it is imperative to use a structured approach that describes how the project will be carried out. The choice of project management is a decisive phase in the successful completion of the project [1]. A working methodology and development process must be defined, and a project schedule drawn up. In this work, based on the results obtained concerning the conceptualization of our learning activity in a pedagogical scenario according to the modular system, and after having defined our choices in terms of development process and modeling language, we propose a modeling with UML of a system for the conceptualization of a learning situation of a model in online teaching for an inductive approach. Finally, we provide examples of models for our system.

Nevertheless, designing scripting tools for learning activities adapted to blended learning is a complex process that requires a coherent, well-thought-out plan to ensure successful learning [2]. This requires consideration of the pedagogical objectives of the activity, the variety of tools and strategies available, and the needs and interests of the learners [3, 4]. It is important to take this aspect into account, especially when considering a diverse group of distance learners, to be able to offer them appropriate motivational strategies in e-learning systems [5]. Scripting tools are tools used to define and create a series of pedagogical activities in an online environment. They comprise activity diagrams and specification tables to help teachers define pedagogical objectives and organize pedagogical activities in a rational way [6, 7]. On the other hand, teachers can use activity diagrams to clarify pedagogical objectives and organize pedagogical activities, which is an excellent way of planning pedagogical activities and defining tasks in diagrammatic form. A datasheet is used to identify the requirements and methods of a task to be performed, and to provide detailed information.

By using both tools, online instructional storyboard designers can approach their scripts in an organized and rational way. These tools can provide an organized approach to the scriptwriting process, which many people find useful when learning. Within the intricate cycle of a pedagogical scenario in a learning situation lies the revelation of "The four Types of Scripts in an Online Learning Situation" [8]. This paints the picture with four unique stages and related activities. Initiation is up first. Presentation lies in presenting content, performing diagnostic evaluations, and delivering remedial backstage support. The next stop on this cycle is Conceptualization wherein learners dive into tasks such as expounding concepts, using deductive or inductive approach, and arranging information, Group discussions, the practical application of knowledge, and continuous assessment - these activities make up the meat of the Objectivation phase. In the last phase, the Transfer Phase, a pivotal stage in the pedagogical framework is reached. During this phase, a detailed examination of case studies takes place, allowing for a comprehensive analysis. Thorough summative assessments serve as a fundamental cornerstone in evaluating the overall learning outcomes. This phase, aptly named the Transfer Phase, sets the stage for the application of acquired knowledge and the cultivation of practical competencies.

Following having observed the stages of the learning cycle outlined hitherto, conceptualization comprises the secondary phase. It entails structuring the progression from action to implementation, founded upon hindrances encountered throughout situational undertakings. In the end, this stage fosters meaning-making surrounding specialized knowledge, its implementation, and integration in skill development. The stage encompasses two scenarios contingent on the adopted approach. Indeed, contingent on context and situation, proposed learning activities necessitate opting between deductive and inductive approaches. Based on this digital pedagogical framework, presently we examine the second phase concerning conceptualization, with particular focus on the inductive approach.

As part of our comprehensive study, we delve into the second phase of conceptualization, especially the enigmatic inductive approach. As part of our research, we first examine a comprehensive theoretical framework, wherein we reveal the essence of the inductive approach, followed by the presentation of a meticulously constructed scenario designed for the conceptualization of a learning situation of a module in online teaching for an inductive approach.

As we progress, we engage in a compelling discourse on the development process and modeling language employed in our study. We embrace the 2TUP process as a symbol of structural integrity and efficiency while harnessing the potential of UML (Unified Modeling Language). Regarding the modeling system, we venture into the static view, showcasing the class diagram and use case diagram. Further into the dynamic view, we present the modeling of activity and sequence diagrams.

The subsequent phase pertains to the prototyping stage, wherein we materialize our findings through visually perceptible manifestations. Within this context, we present a series of five figures that serve as graphical representations of our system. The initial illustration portrays the authentication model, followed by compelling overviews of the learner dashboard and the teacher dashboard. Additionally, we unveil mock-ups of the wiki and chat pages, thoughtfully presented in both light and dark modes.

II. Theoretical Framework

A. *Scenario for the Conceptualization of a Module Learning Situation in e-Learning for an Inductive Approach*

The inductive approach necessitates proceedings from the explicit to the generalized study in [9]. It represents a scientific technique for deriving broad conclusions from individual premises. This enables learners to discover conceptual significance throughout learning undertakings experimentally. It permits transitioning from actual or concrete observations and examinations to more generalized perspectives. Generalization (objectivization) encourages learners to depict applied methodologies and designate involved operations, drawing upon metacognition, while prompting them to critically and holistically examine their knowledge structures [10]. This process of interdisciplinarity catalyzes the cultivation and reliance on meta-cognitive skills. It fosters a coherent merging of all targets attained, employing dialogue to compile and summarize outcomes. Through generalization, learners explicate individualized scholarship techniques and procedures in aim-related terminologies that can be comprehended and applied more universally [11]. Fig. 1 exhibits an instance of an activity scheme conceptualized for an inductive pedagogical approach concerning a learning scenario founded on the modular system, composed of three systems: input system, Learning system, and output system.

The proposed model is interpreted according to its three systems as follows:

The input system for inductive conceptualization scenarios presents learning activities through defining objectives, knowledge, and skills learners must master.

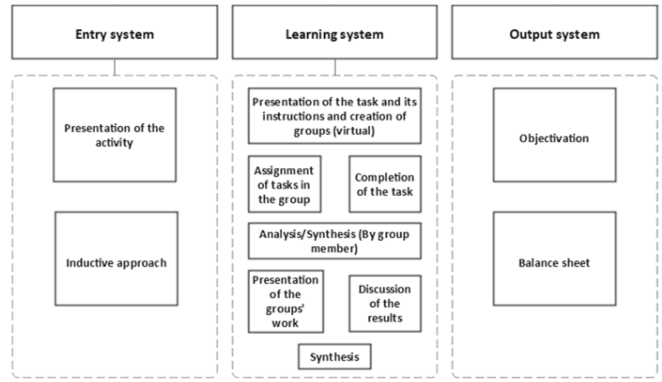

Fig. 1. Example of a scenario for conceptualizing an inductive approach to a learning situation.

The learning system of the conceptualization activity scenario is based on an inductive approach, involving the transition from specific to general principles [9]. It is consequently prudent to employ the inductive approach, concurring with learner-focused oblique instruction strategies. Indeed, queries, induction, problem resolution, judgment formulation, and discovery are terms interchangeably used to illustrate oblique pedagogy [12, 13]. Indirect teaching advances resourcefulness and interpersonal ability progress. Our system originally proffers a presentation of the undertaking to be fulfilled, contingent on the idea's attributes to be addressed (experiment, problem to be solved, investigation, etc.).

The subsequent step includes enacting the proposed undertaking relative to learner parties' characterizations (autonomous or collaborative exertion), transiting through diverse steps: undertaking accomplishment by organizing and allotting duties in collaborative circumstances, without neglecting time direction; offering explicit experimental constituents for examination by stimulating interrogatives, trials, manipulations and hypothesis development, as well as supplying methods and materials; and ultimately provide feedback by endorsing triumphs, enhancements and self-reformation mediums along with assisting error-discrimination and application. It must be noted undertaking finalization demands supplemental digital asset provision to support learners' labor, and communicative technologies to facilitate interplay between various instructors and learners.

The final step of this educational process involves the examination of the outcomes produced by learners or groups of learners utilizing different communication technologies. Initially, presenting and sharing the results of each individual or group with all participants (teachers and learners) is essential. Following this, the results must be interpreted to validate the empirical findings and compare them with theoretical principles and regulations.

The output system of the scenario of the conceptualization activity of an inductive approach concerns an evaluation of said conceptualization activity as proposed by generalization through engaging metacognition via the manifestation of a model or principles utilizing mathematical formalism.

B. Development Process and Modeling Language

The development process is a decisive factor in the success of a project [14], which outlines the project's phases and determines its fundamental characteristics. Therefore, selecting a development method that is suitable for the project's specific needs and requirements is essential to create a high-quality project that satisfies users' expectations. There are various unified processes, including Extreme Programming (XP) and Rational Unified Process (RUP) [15]. In this particular project, the chosen method is the 2TUP process, which is a hybrid development model that combines the strengths of both RUP and XP and integrates their respective approaches. Our project required a unique approach, and we ultimately decided to utilize the 2TUP process. This process combines the best aspects of both RUP and XP, resulting in a hybrid software development model that integrates their respective approaches. To aid in our project, we employed the unified two-track process, also known as the Y development model. This model follows a split approach, where requirements are studied in one track and technical aspects in the other, before merging them in the lower branch [16]. The Y development model is versatile and suitable for projects of all sizes; it plays a crucial role in managing technical risks and project domains. Additionally, the model addresses the challenges of continuous evolution in information systems by breaking down system analysis along functional and technical axes [17].

Modeling is the design of models. Contingent upon intention and mediums applied. Within a computational discipline, the specification of a phase in constructed informational systems is denoted through data modeling [18]. UML comprises validated engineering optimum processes for modeling sizable, sophisticated schemes [19]. The 2TUP operation relies on UML (Unified Modeling Language) throughout development cycles [20], since its diverse schemas facilitate then simplify the correct modeling of the system at all phases. Verily, UML is depicted as a graphical and textual modeling communicative contrived to apprehend and portray requisites, stipulate, blueprint remedies, and transmit notions [21].

UML unifies object-oriented notation and concepts. It is not alone a representation, but the ideas passed on by the diagrams hold accurate semantics which means in an equivalent approach as text in a language, therefore UML is sometimes depicted as a technique when it truly isn't. UML can be utilized to produce a variety of representations to aid hardware design and development. UML diagrams in use include use case diagrams, class diagrams, sequence diagrams, and activity diagrams [22]. It has two factors: static modeling, which addresses method structure, and dynamic modeling, which concerns the system behavior. Use case diagrams and class diagrams are used to build the basis for static modeling. Dynamic modeling, contrarily, is based on activity and sequence diagrams [23].

UML also unifies the notation required for the various activities in the development process, and hence provides a method for subsequent decision-making, from requirements definition to coding [17]. It is the result of the unification of mature techniques for the analysis and design of large software packages and complex systems.

Consequently, we can conclude that the UML language helps us in all phases of the project, as it offers many advantages in system analysis and design. Consequently, a method is proposed combining UML and Unified Process to guide the realization of object-oriented systems.

III. System Modeling

Given the theoretical data concerning the stages involved in creating an instructional scenario for learning in various disciplines, and more specifically in the case of an online instructional scenario, we analyzed the various components of instructional design for the creation of scenario tools. We based our analysis on the 2TUP development process and the UML modeling language. To create our diagrams, we used StarUML software [24], which also enabled us to make rapid modifications and adjustments to our diagrams.

A. Static View

Static UML (Unified Modeling Language) diagrams are used to model the static structure, i.e., its composition in terms of objects, classes, packages, and so on. There are two diagrams, the class diagram and the use case diagram.

The class diagram is used to represent classes, their attributes, methods, relationships (association, aggregation, composition, inheritance, etc.) a constraint [25].

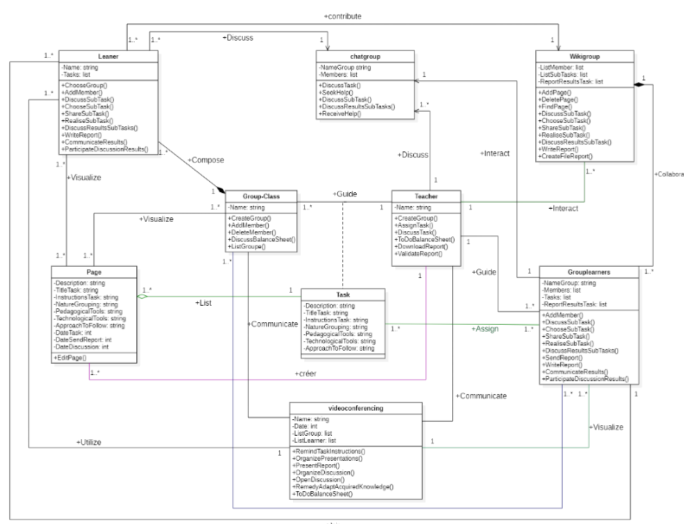

Fig. 2. Class diagram.

- In Fig. 2, which represents the class diagram, we have represented eight classes and one associative class. The eight classes are as follows: Teacher, class-group, learner, group chat, group wiki, learner group, videoconference, and page. The associative class is "task", which results from the relationship between the teacher and the class-group.
- The "Teacher" class enables a teacher to manage groups and assigned tasks. It contains operations such as "create_group()" to create a new workgroup,

"assign_tasks()" to assign tasks to a specific group, "download_reports()" to retrieve learners' reports, and "make_assessment()" to evaluate the group's performance.

- The "Classgroup" class is used to manage workgroups. It contains operations such as "add_member()" to add a member to an existing group, "delete_member()" to remove a member from the group, and "list_group()" to display group members. For example, a group or teacher can use this class to add new members to the group, delete existing members, and display the list of current members.

- The "Task" class is an associative class representing a task assigned to a specific group. It contains attributes such as "description", "title", "TaskInstructions", "GroupNature", "PedagogicalTools", "TechnologicalTools" and "ProcedureToFollow". For example, a task must include a description of the functionality to be implemented, instructions for setting up the functionality, tools, and techniques to be used, and a procedure to be followed to achieve the objective.

- The "Learner" class represents an individual learner. It contains operations such as "choose_group()", "add_member()", "realize_under_task()", and "write_report()", which enable the learner to engage in the learning process. For example, a learner can use this class to choose a specific workgroup, carry out assigned tasks, and write reports on the results obtained.

- The "LearnerGroup" class represents a group of learners. It contains operations similar to those of the "Learner" class, except for an entire group. For example, a group of learners can use this class to discuss assigned tasks, perform tasks together, and communicate results.

- The "Page" class represents a web page containing information about an assigned task. It contains attributes such as "TaskDate", "ReportDate" and "DiscussionDate", along with an "edit_page()" operation for modifying the page. For example, a page can contain information on the functionalities to be implemented, the tools to be used, the deadlines to be met, and the resources available.

- The "Videoconference" class can be used to set up videoconferences to discuss assigned tasks and review progress. It contains operations such as "remind_tache_instructions()", "organize_presentations()", and "make_review()". For example, a group can use this class to organize a videoconference to discuss the tasks assigned, present the results obtained, and evaluate the group's performance by the teacher.

- The "GroupChat" class enables group members to discuss assigned tasks and request assistance. It contains operations such as "discuss_Task()", "request_help()", and "receive_help()". For example, a group can use this class to discuss with the teacher a problem encountered during the completion of a task.

- The "WikiGroupe" class lets you create and manage a wiki for a workgroup. It contains operations such as "add_Page()", "delete_Page()", "find_Page()", "discuss_subtache()", "choose_subtache()", "share_subtache()", "make_subtache()", "discuss_results_subtache()", and "create_report_file()". For example, a group needs to use this class to create a wiki dedicated to their task, add pages for each subtask, discuss subtasks, share useful files and links, and write reports on the results achieved.

Use case diagram represents the interactions between a system and its actors (users, other systems, etc.) [26] in terms of use scenarios and functionalities offered by the system. The use case diagram is often used at the start of the design phase to identify user needs and the main functionalities to be developed. In a use case diagram, actors are represented as external blocks and use cases as ellipses. Relationships between actors and use cases are represented by arrows.

Per the use case diagram shown in Fig. 3, the teacher introduces the activity and corresponding task, forms learner groups, assigns tasks, and oversees assessment and remediation procedures. He assists in group formation and facilitates assessment discussions for the entire class. Within learner groups, participants work collaboratively on the task, delegate sub-tasks, and review and summarize the results. The work is submitted to the educator before the presentation, as per the "include" use case relationship where Case B only transpires upon the execution of Case A.

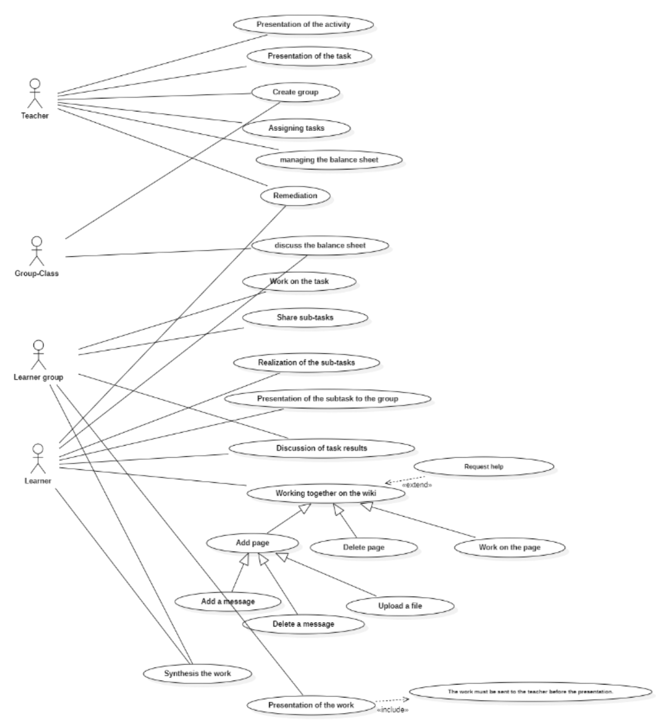

Fig. 3. Use case diagram.

Ultimately, the learner assumes responsibility for conducting sub-tasks within the group, engaging in discussions, collaborating on the wiki platform, and seeking guidance from the teacher if needed, in line with the "extend" use case relationship where Case B can be extended through Case A execution. It is important to note that use case associations merely illustrate interactions between actors and scenarios, not prescribing an accurate sequential order or information flow [27].

B. Dynamic View

UML dynamic diagrams are used to model dynamic behavior, i.e., responses to events and stimuli. In our case, we use the following diagrams sequence and activity diagrams

An activity diagram represents the flow of activities in a process or procedure. It models decision-making, loops, synchronization, and operations specific to task processing.

As shown in Fig. 4, we begin by placing a start node at the center of our activity diagram; this symbolizes the beginning of the activity and is represented by a black circle. This is where the teacher introduces the activity and presents the task and instructions to the group/class. From here, groups are created by the teacher, and the group/class is formed.

Once the task has been assigned to the group, learners can begin to carry it out. If they manage to complete the task successfully, the group of learners starts writing and sending reports. If, on the other hand, the group encounters difficulties, the teacher assists.

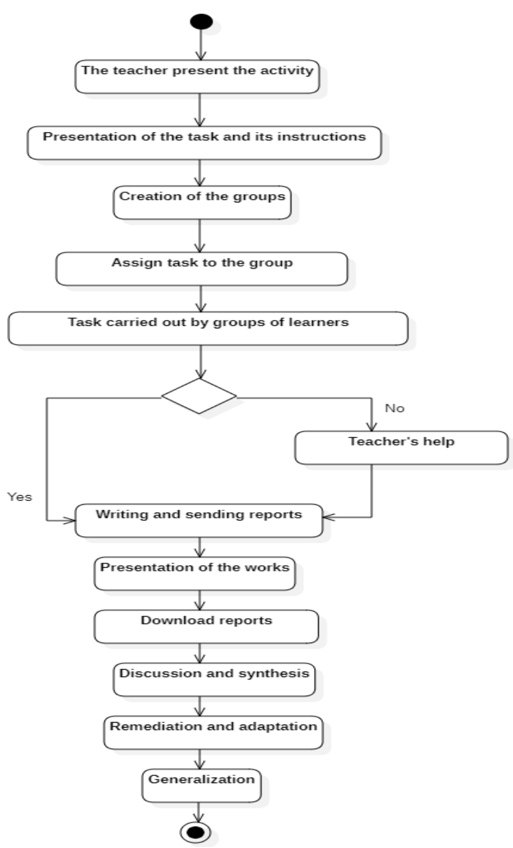

Fig. 4. Activity diagram.

After the teacher has uploaded the reports for validation, the groups present their work. Discussion and synthesis then follow, and the teacher can make any necessary adjustments. Finally, the generalization concludes the activity, allowing the teacher to recap the objectives of the activity and encourage learners to continue to work hard and persevere in their learning. The final knot is represented by an outlined black circle.

The sequence diagram shows the interaction between objects over time. It visualizes the sequence of messages exchanged between objects to carry out a given task for a given scenario.

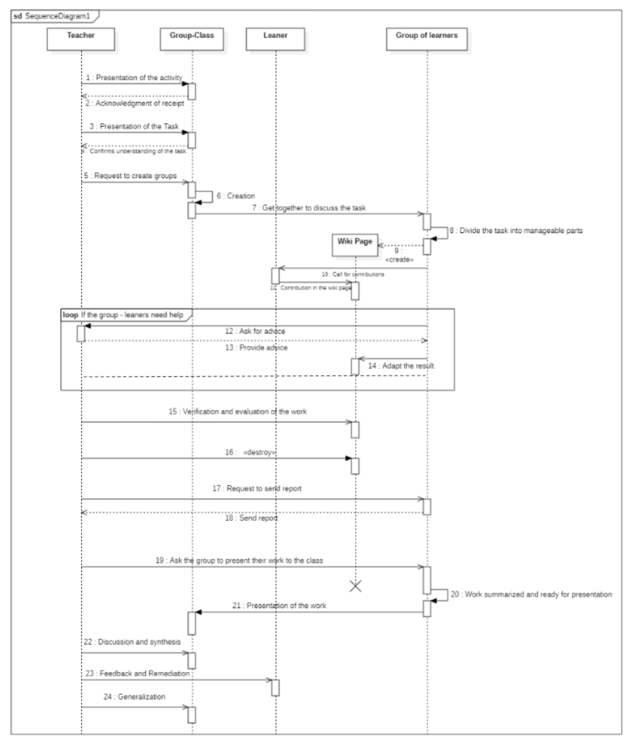

Fig. 5. Sequence diagram.

The sequence diagram illustrates the interactions between different actors and the main functions of a system as shown in Fig. 5. It shows how the different actors interact to accomplish a task. It starts with the teacher providing knowledge and tasks to the learners and then presenting the task and instructions to the learners. It is inevitable to ask for the creation of groups so that learners can collaborate.

Each member can contribute information to the wiki to contribute to the task.

Our sequence diagram also includes a loop that allows learners to get help from the teacher if they encounter difficulties. By incorporating the teacher's feedback, the group of learners can modify their results to suit their needs. Once the learners have completed their work, the teacher checks and evaluates it. Once the evaluation is complete, the teacher initiates the next phase, requesting reports for the class group. The results of the learners' work can then be discussed and summarized with the teacher. In the event of gaps, the teacher can provide feedback and suggest remedial action if necessary.

In brief, of the UML modeling of our system, once the crucial details had been established, we concentrated on the UML modeling of the system. We created a total of four diagrams, two of them static. The first is the class diagram, made up of eight classes, including an associative class called "task". The second diagram is the use case diagram, which links four actors to fifteen primary tasks and five secondary tasks.

As far as the dynamic diagrams used are concerned, two diagrams were used in this case. The activity diagram was used to visually demonstrate the progression of various actions within a specific process or procedure. In our particular case, we chose to illustrate the progression of the inductive approach in a learning scenario. The second diagram used is the sequence diagram, which delineates the distinct interactions between the various actors involved in the process.

IV. PROTOTYPING

Human-machine interfaces and computer ergonomics are essential elements of modern technology, and can significantly improve user experience and productivity. User-centered design (UCD) is a methodology that prioritizes user needs and preferences in the design process, to create products that are intuitive, efficient, and enjoyable to use [28]. In addition, the use of prototypes will help teams brainstorm and organize their thoughts, leading to more effective problem-solving and collaboration [29]. On the other hand, a functional model or wireframe refers to a diagram used to define the areas and components of a user interface during its design. A wireframe can be created using various methods, such as sketching, paper collage, or digital diagrams.

In other words, wireframe modeling is a method used in user experience (UX) design, wireframe modeling enables us to: Identify and address usability issues early in the design process, such as layout. Navigation and content organization. In addition, it allows us to identify potential conflicts between user needs and application capabilities, as well as gaps in the user interface.

The examples in this section illustrate the proposed models. Adobe XD is used to implement these models. Through this work, we propose five models.

A. Authentication Mock-up

The authentication interface enables learners to connect to their workspace using their student ID or institutional address. This provides learners with a simple and secure means of accessing their workspace and collaborating with their peers. Fig. 6 shows a mock-up of the authentication interface enabling users to access the platform.

The username field is designed to accept the learner's ID number or institutional e-mail address, which is linked to their profile. Once the user has entered their credentials, they can click on the validation button to access their workspace. The password field is designed to guarantee the security and confidentiality of the user's account. The password is encrypted and securely stored, and the user is prompted to enter it each time he or she accesses the platform. This is an additional level of security that prevents unauthorized access to the user's workspace. The logo on the authentication interface serves as a visual identifier for the institution and adds a professional touch to the interface. It also reinforces the institution's brand identity and creates a sense of familiarity for users.

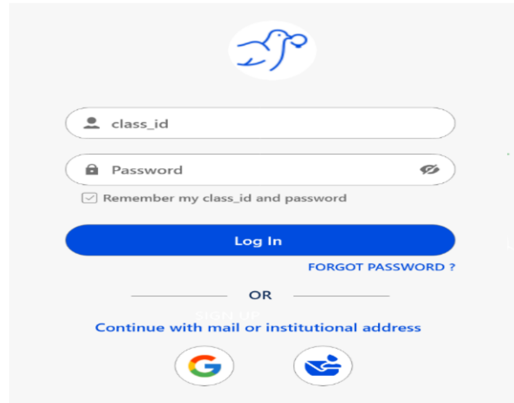

Fig. 6. Login page.

B. Mock-up for the Dashboard

The dashboard displays the various tasks and activities that have been assigned by teachers, along with the corresponding due dates and deadlines. In addition, learners can see the total time they have spent on the platform, helping them to manage their time effectively. Fig. 7 shows the learner dashboard, which provides an overview of their progress through the course and enables them to monitor their completion status.

Fig. 7. Learner dashboard.

The dashboard also includes a calendar that allows learners to see the schedule of upcoming sessions and plan their work accordingly. This feature helps learners stay organized and ensure they don't miss any important sessions or deadlines. In addition, the learner dashboard includes a communication function that allows learners to contact their teachers directly. This function enables learners to ask questions, seek clarification, or provide feedback on course material or teaching. This direct communication channel helps foster a collaborative learning environment and ensures that learners receive the support they need to succeed in the course. In summary, the learner dashboard in Fig. 7 provides an overview of learners' progress in the course, allowing them to track their completion status, view upcoming sessions, communicate with teachers, and access personalized learning resources. The dashboard is designed to be user-friendly and intuitive, making it easy for learners to navigate and use.

C. Mock-up for the Teacher Dashboard

The dashboard is designed to be intuitive and user-friendly, giving teachers easy access to the information they need to teach and manage their courses effectively. Fig. 8 shows the teacher dashboard, which serves as a hub for managing courses, monitoring learners' progress, and communicating with them.

At the head of the dashboard, the teacher can see a planning section for the next session. This section displays the date, time, and duration of the upcoming session, as well as a list of tasks and activities to be covered. The teacher can use this section to prepare for the upcoming session, ensuring that all necessary materials and resources are available. Under the planning section, the teacher can see a folder in which to upload files for each class. This folder enables the teacher to store and organize resources, such as lesson plans, slides, and handouts, in one convenient place. The teacher can also see the progress of the class, including which learners have accessed the material and what tasks have been completed.

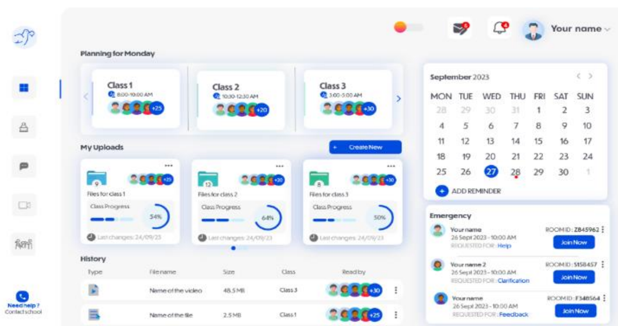

Fig. 8. Teacher dashboard.

In addition, the teacher can consult an emergency section on the dashboard. This section displays a list of learners who need help or feedback on their work. The teacher can quickly identify learners who need help and provide them with the necessary support. At the top of the dashboard, the teacher can also see incoming messages or notifications. This enables them to keep abreast of important announcements or messages from learners, parents, or other teachers. Finally, the teacher can change the look and feel of the dashboard by choosing between a light and dark mode. In this way, teachers can customize their dashboard to suit their personal preferences and working style.

D. Mock-up of Wiki and Chat Pages

The wiki page allows learners to collaborate and share information, files, and resources, enabling them to work effectively together to achieve a common goal. Fig. 9 illustrates a collaborative learning environment in which a group of learners work together to complete a task proposed by a teacher.

Learners use a wiki page to organize their work, adding and modifying the pages needed to complete the task. On the right-hand side of the interface, learners can access a chat function that enables them to communicate with each other in real-time. This feature enables learners to ask questions, request help, and provide feedback to each other, fostering a collaborative and positive learning environment.

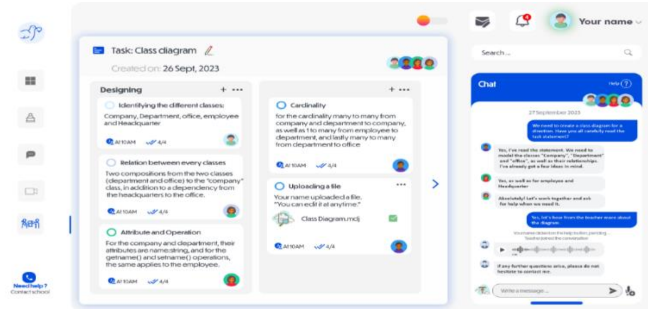

Fig. 9. WikiPage et chat.

The wiki page also allows learners to upload files, such as documents, images, and videos, to support their work. Learners can add new pages to the wiki as required, and they can modify or delete existing pages as necessary. This flexibility enables learners to organize their work in the way that best suits their needs, and promotes effective collaboration.

In one of the many features of the interface, in this case, the chat function, learners can ask the teacher or other experts for help. Learners can use this function to ask questions, request feedback, or seek clarification on any aspect of the task. The teacher or other experts can then respond to learners' requests, providing advice and support where necessary.

E. Mock-up of Dark Mode

Dark mode is often used to reduce eyestrain [30] and preserve battery life on devices with OLED displays [31], as well as to offer users a more cinematic and immersive experience. Dark mode can also be aesthetically pleasing, with many users finding it elegant and modern. In low-light environments, dark mode can also be easier on the eyes, as it reduces the amount of blue light emitted by the screen. Fig. 10 shows an example of dark mode, a display setting that inverts the color palette of a user interface, with a dark background and light text.

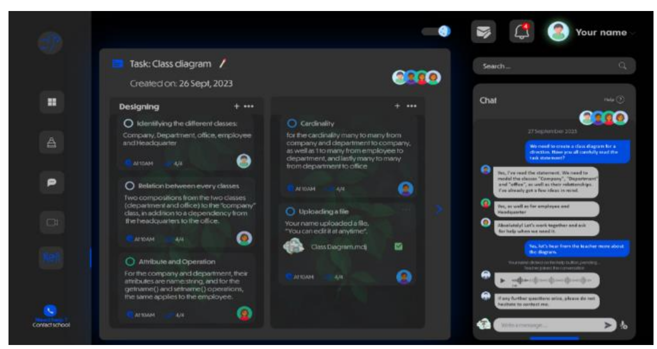

Fig. 10. Dark mode.

To bring this section to a close, we've illustrated five mock-ups. The prototype showcases the transparent authentication process, designed to guarantee secure access to our platform. Users will appreciate the elegant, intuitive interface that simplifies the login procedure.

The second and third prototypes highlight the learner and teacher dashboards, respectively. These well-designed interfaces provide users with an overview of their progress, performance, and personalized learning paths.

Finally, we presented the last two prototypes, which feature a wiki function with chat, a collaborative space for knowledge sharing and discussion. In both prototypes, users can create and edit content, take part in lively discussions, and seek clarification from their peers. Yet the second prototype goes a step further by integrating a visually appealing dark mode option. This feature not only enhances the overall aesthetic experience but also meets the needs of users who prefer a more discreet, immersive interface.

V. Discussion and Limitations

This study proposed a conceptual model for designing inductive learning activities in an online teaching scenario. By grounding the model in established pedagogical frameworks such as the four stages of the learning cycle [8] it aimed to provide a coherent and structured approach.

We applied a structured systems development approach, as previous research has highlighted the importance of structure in projects [32]. As part of our effort to enhance the efficiency and organization of our work, defining a methodology, development process, and schedule was an essential step [33].

We explored conceptualizing an online inductive learning situation using the Unified Modeling Language (UML) for modeling. UML provides a standardized way to visually represent a system's structure and behavior through diagrams like class, use case, activity, and sequence when applied according to established standards [27]. These visual representations, if properly executed facilitate effective communication and collaboration among stakeholders by enabling a shared understanding of system components and functionality [34].

We adopted the 2TUP process to ensure our project followed a systematic, well-defined approach consistent with recommended practices. As described in the relevant literature, 2TUP is rooted in UML and advocates integrating this modeling language throughout the entire development cycle [32]. When applied correctly, UML lends simplicity and clarity to diagrams as an effective means of representation.

By applying this structured systems development approach to analysis and design, we aimed to fulfill system requirements by developing models using object-oriented design and relational database techniques [34]. Creating static and dynamic models allowed us to better understand user requirements and design user-centered solutions.

The prototype we showed demonstrated potential interface screens to refine before full implementation, reflecting best practices. [35]. Learner dashboards and wikis, in particular, depicted collaborative, project-based activities aligned with methods shown by relevant research to motivate e-learners.

In terms of supporting the inductive approach highlighted in existing pedagogical frameworks, the system allows learners to engage with examples and tasks before higher-level concepts are defined. This mirrors the inductive process of drawing inferences from specific cases. Dashboards, wikis, and chat provide a blended environment for facilitating exploration, discussion, and progressive sense-making known from the literature to develop a deeper understanding versus traditional deductive methods [36].

The conceptual models we developed help visualize and plan an online inductive learning scenario according to established pedagogical frameworks. By applying UML and 2TUP, as advocated in relevant sources, we aimed to create a coherent system addressing real user needs, with static and dynamic diagrams clearly illustrating key entities, interactions, and workflows. This type of modeling approach facilitates explaining complex conceptual systems in an organized, visual manner.

The main objective of this article is to discuss the stage of conceptualization, with a specific emphasis on the inductive approach within the pedagogical framework. Although our research delves into the intricacies of this stage, it is important to acknowledge that our focus is limited and does not encompass the entirety of the pedagogical framework. This framework encompasses various other stages, such as the deductive approach, evaluation methods, and other crucial components. While our work is valuable in exploring the inductive approach, it should be viewed as a starting point for further research to incorporate these essential elements. By recognizing these limitations, we can pave the way for future studies that take a more comprehensive and holistic approach to online learning.

VI. Conclusion

In short, our work highlights the fundamental importance of a structured, methodical approach to IT projects, focusing specifically on the design of e-learning pedagogical scenarios for the inductive approach.

After careful evaluation, we decided to adopt the 2TUP (2-Track Unified Process). This approach is well-suited to projects of all sizes and enables us to effectively manage the ongoing evolution of information systems. The division of system analysis into functional and technical aspects offered by 2TUP enabled us to better manage technical risks and ensure effective project management.

We devoted our efforts to UML modeling of our system, creating four diagrams, including two static diagrams. The class diagram presents eight classes, including an associative class called "task". The use case diagram links four actors with fifteen main tasks and five sub-tasks. In contrast, we used two dynamic diagrams: the activity diagram, which illustrates the flow of actions in a process, and the sequence diagram, which describes the interactions between the actors involved.

We aimed to conceptualize an inductive learning situation, with a focus on e-learning. We used UML (Unified Modeling Language) to graphically represent the various components of the system. In turn, we created five concrete mock-up examples to illustrate the system's operation and appearance practically.

In the next stage, our system will undergo a complete development process. It will be developed and rigorously tested to ensure its readiness for deployment. An evaluation will be carried out to measure its effectiveness. Finally, the system will be deployed to its target audience.

REFERENCES

[1] HYVÄRI, Irja. Project management effectiveness in project-oriented business organizations. International journal of project management, 2006, vol. 24, no 3, p. 216-225.

[2] VAUGHAN, Norman D., CLEVELAND-INNES, Martha, et GARRISON, D. Randy. Teaching in blended learning environments: Creating and sustaining communities of inquiry. Athabasca University Press, 2013.

[3] KAHAN, Tali, SOFFER, Tal, et NACHMIAS, Rafi. Types of participant behavior in a massive open online course. International Review of Research in Open and Distributed Learning, 2017, vol. 18, no 6, p. 1-18.

[4] PARK, Ji-Hye et CHOI, Hee Jun. Factors influencing adult learners' decision to drop out or persist in online learning. Journal of Educational Technology & Society, 2009, vol. 12, no 4, p. 207-217.

[5] BOVERMANN, Klaudia et BASTIAENS, Theo J. Towards a motivational design? Connecting gamification user types and online learning activities. Research and Practice in Technology Enhanced Learning, 2020, vol. 15, no 1, p. 1-18.

[6] ANOIR, Lamya, KHALDI, Mohamed, et ERRADI, Mohamed. Personalization in Adaptive E-Learning. In: Designing User Interfaces With a Data Science Approach. IGI Global, 2022. p. 40-67.

[7] KAWTAR, Zargane, MOHAMED, Erradi, et MOHAMED, Khaldi. E-learning adaptive and the tools of screenwriting: a case of collaboration. Global Journal of Engineering and Technology Advances, 2021, vol. 7, no 3, p. 203-212.

[8] MAHA, Khaldi, OMAR, Erradi, MOHAMED, Erradi, et al. Design of educational scenarios of activities in a learning situation for online teaching. GSC Advanced Engineering and Technology, 2021, vol. 1, no 1, p. 049-064.

[9] VARPIO, Lara, PARADIS, Elise, UIJTDEHAAGE, Sebastian, et al. The distinctions between theory, theoretical framework, and conceptual framework. Academic Medicine, 2020, vol. 95, no 7, p. 989-994.

[10] GODDIKSEN, Mads et ANDERSEN, Hanne. Expertise in interdisciplinary science and education. 2014.

[11] MCKEOUGH, Anne, LUPART, Judy Lee, et MARINI, Anthony (ed.). Teaching for transfer: Fostering generalization in learning. Routledge, 2013.

[12] BRUNER, Jerome S. The process of education. Harvard university press, 2009.

[13] JO, Hyun-Jae, CHUNG, Seung-jin, et SATTERFIELD, Debra. Indirect Teaching for All and Autism Spectrum Disorder (ASD) in Design Class (ITAD) Encouraging Their Emotional Empathy. In : Advances in Affective and Pleasurable Design: Proceedings of the AHFE 2016 International Conference on Affective and Pleasurable Design, July 27-31, 2016, Walt Disney World®, Florida, USA. Springer International Publishing, 2017. p. 581-591.

[14] GOVIL, Nikhil et SHARMA, Ashish. Estimation of cost and development effort in Scrum-based software projects considering dimensional success factors. Advances in Engineering Software, 2022, vol. 172, p. 103209.

[15] Sharma, N., & Wadhwa, M. Exsrup: Hybrid software development model integrating extreme programing, scrum & rational unified process. TELKOMNIKA Indonesian Journal of Electrical Engineering, 2015, 16(2), 377-388.

[16] Céret, E., Dupuy-Chessa, S., Calvary, G., Front, A., & Rieu, D. A taxonomy of design methods process models. Information and Software Technology, 2013, 55(5), 795-821.

[17] P. Roques, F. Vallée, "UML 2 in action - From needs analysis to J2EE design," Eyrolles, pp. 382, March 2007. ISBN: 978-2-212-12104-9.

[18] VARENNE Franck, SILBERSTEIN Marc, DUTREUIL Sébastien et al., Modeling & simulating – Volume 2. Epistemologies and practices of modeling and simulation. Éditions Matériologique, "Modelling, simulations, complex systems", 2014, ISBN: 9782919694730. DOI: 10.3917/edmat.varen.2014.01. URL: https://www.cairn.info/modeliser-et-simuler--9782919694730.htm

[19] Weilkiens. Systems engineering with SysML/UML: modeling, analysis, design. Elsevier, 2011.

[20] RAMDA, Amel, HARRAK, Sihem, and BOUAZIZ, Hamida Framer. Development of a Mobile Application for Timetable Management at the University of Jijel. 2019. Doctoral dissertation. Jijel University.

[21] CHARROUX, Benoît, OSMANI, Aomar, and THIERRY-MIEG, Yann. UML 2: modeling practice. Paris: Pearson Education, 2010.

[22] LIU, Xianhong. Identification and check of inconsistencies between UML diagrams. In : 2013 International Conference on Computer Sciences and Applications. IEEE, 2013. p. 487-490.

[23] KHALDI, Maha et ERRADI, Mohamed. Design and Development of an e-Learning Project Management System: Modelling and Prototyping. International Journal of Emerging Technologies in Learning (iJET), 2020, vol. 15, no 19, p. 95-106.

[24] StarUML. (n.d.). StarUML. Retrieved October 14, 2023, from https://staruml.io

[25] ANWAR, Muhammad Waseem, RASHID, Muhammad, AZAM, Farooque, et al. A unified model-based framework for the simplified execution of static and dynamic assertion-based verification. IEEE Access, 2020, vol. 8, p. 104407-104431.

[26] ALERYANI, Arwa Y. Comparative study between data flow diagram and use case diagram. International Journal of Scientific and Research Publications, 2016, vol. 6, no 3, p. 124-126.

[27] ADNAN, Nor Hafizah et RITZHAUPT, Albert D. Software engineering design principles applied to instructional design: What can we learn from our sister discipline?. TechTrends, 2018, vol. 62, p. 77-94.

[28] LOWDERMILK, Travis. User-centered design: a developer's guide to building user-friendly applications. " O'Reilly Media, Inc.", 2013.

[29] DEININGER, Michael, DALY, Shanna R., SIENKO, Kathleen H., et al. Novice designers' use of prototypes in engineering design. Design studies, 2017, vol. 51, p. 25-65.

[30] ERICKSON, Austin, KIM, Kangsoo, BRUDER, Gerd, et al. Effects of dark mode graphics on visual acuity and fatigue with virtual reality head-mounted displays. In : 2020 IEEE Conference on virtual reality and 3D user interfaces (VR). IEEE, 2020. p. 434-442.

[31] EISFELD, Henriette et KRISTALLOVICH, Felix. The rise of dark mode: A qualitative study of an emerging user interface design trend. 2020.

[32] KERZNER, Harold. Project management: a systems approach to planning, scheduling, and controlling. John Wiley & Sons, 2017.

[33] KHALDI, Maha, ANOIR, Lamya, ERRADI, Omar, et al. From Analysis to Development of the E-Learning Project Management System (SGPE). In : Handbook of Research on Scripting, Media Coverage, and Implementation of E-Learning Training in LMS Platforms. IGI Global, 2023. p. 94-132.

[34] CAVIQUE, Luis, CAVIQUE, Mariana, MENDES, Armando, et al. Improving information system design: Using UML and axiomatic design. Computers in Industry, 2022, vol. 135, p. 103569.

[35] Andreasen, M M, C Thorp Hansen, and P Cash. *Conceptual Design: Interpretations, Mindset and Models*. *Conceptual Design: Interpretations, Mindset and Models*. Springer. 2015. https://doi.org/10.1007/978-3-319-19839-2

[36] BENITEZ-CORREA, Carmen, GONZALEZ-TORRES, Paul, et VARGAS-SARITAMA, Alba. A Comparison between Deductive and Inductive Approaches for Teaching EFL Grammar to High School Students. International Journal of Instruction, 2019, vol. 12, no 1, p. 225-236.