ΜΑΘΗΣΙΑΚΗ ΠΛΑΤΦΟΡΜΑ ΜΕ ΔΥΝΑΤΟΤΗΤΑ ΕΞΑΤΟΜΙΚΕΥΜΕΝΟΥ
ΠΕΡΙΕΧΟΜΕΝΟΥ

του

ΧΑΡΑΛΑΜΠΟΣ ΚΡΕΜΜΥΔΑΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

υποβληθείσα σε μερική εκπλήρωση των απαιτήσεων για το πτυχίο

ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΕΛΛΗΝΙΚΟ ΜΕΣΟΓΕΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

2023

Εγκεκριμένη από:

Δρ. Βιδάκης Νικόλαος

LEARNING PLATORM WITH PERSONALIZED CONTENT

by

CHARALAMPOS KREMMYDAS

A THESIS

submitted in partial fulfillment of the requirements for the degree

BACHELOR OF INFORMATICS ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

SCHOOL OF ENGINEERING

HELLENIC MEDITERRANEAN UNIVERSITY

2023

Approved by:

Dr. Vidakis Nikolaos

Περιγραφή

Οι πλατφόρμες ηλεκτρονικής μάθησης έχουν αυξηθεί σε δημοτικότητα πρόσφατα, προσφέροντας στους ανθρώπους έναν βολικό και αποτελεσματικό τρόπο να μαθαίνουν νέα πράγματα και να βελτιώνουν τα υπάρχοντα. Ωστόσο, δεν είναι όλες οι πλατφόρμες το ίδιο ποιοτικές και ορισμένες έχουν UI που δεν έχουν σχεδιαστεί σωστά. Μια μέτρια ή κακή διεπαφή χρήστη μπορεί να βλάψει σημαντικά την εμπειρία του χρήστη, καθιστώντας δύσκολη την πλοήγηση στην πλατφόρμα, την πρόσβαση σε σχετικό περιεχόμενο και την ενεργή συμμετοχή στη διαδικασία εκμάθησης. Κατά συνέπεια, οι χρήστες μπορεί να απογοητευτούν, να αδιαφορήσουν ή ακόμα και να εγκαταλείψουν την πλατφόρμα εντελώς. Αναγνωρίζοντας τη σημασία ενός καλά σχεδιασμένου UI, θα παρουσιάσουμε μια πλατφόρμα ηλεκτρονικής μάθησης που διαθέτει ανώτερη διεπαφή χρήστη, προσεκτικά επιλεγμένες παλέτες χρωμάτων και φιλική προς το χρήστη πλοήγηση. Στόχος μας είναι να διασφαλίσουμε μια θετική εμπειρία χρήστη για τους μαθητές, προωθώντας τη δέσμευσή τους, την ικανοποίησή τους και, τελικά, την επιτυχία τους στην επίτευξη των μαθησιακών τους στόχων.

Λέξεις κλειδιά: Διεπαφή χρήστη, Εμπειρία χρήστη, UI, UX, Διαδικτυακή μάθηση, αλληλεπίδραση χρήστη-υπολογιστή (HCI)

Abstract

E-learning platforms have grown in popularity recently, offering people a convenient and effective way to learn new things and improve existing ones. But not all platforms are made equal, and some have UIs that are poorly designed. A subpar UI can significantly impair the user experience, making it difficult for learners to navigate the platform, access relevant content, and actively engage in the learning process. Consequently, users may become frustrated, disinterested, or even abandon the platform altogether. Recognizing the importance of a well-designed UI, we will introduce an e-learning platform that features a superior user interface, carefully curated color palettes, and user-friendly navigation. Our aim is to ensure a positive user experience for students, promoting their engagement, satisfaction, and ultimately, their success in achieving their learning objectives.

Keywords: User Interface, User experience, UI, UX, E-Learning, human-computer interaction (HCI)

Acknowledgments

Table of Contents

List of Figures

Chapter 1 - Introduction

E-learning has grown significantly over the past ten years, revolutionizing how people learn new information and acquire new skills. E-learning's rise can be linked to technological advancements, internet accessibility, and rising digital literacy. The global use of e-learning was hastened by the COVID-19 pandemic, which started in late 2019, when traditional teaching techniques became more dangerous. In order to stop the virus from spreading while preserving academic and professional continuity, both businesses and educational institutions were obliged to reconsider their approaches to teaching and training. The greatest strategy for maintaining academic continuity and limiting pandemic spread was to adopt e-learning [1].

With that said, there are a number of difficulties unique to this creative approach to schooling. Maintaining student motivation, ensuring digital equity, offering efficient support, and addressing challenges with user interface design are some of the most urgent problems. In order to fully realize the promise of e-learning and provide world-wide learners with high-quality educational experiences, it is imperative to solve these issues as it continues to develop [2].

1.1 Objectives

In order to improve the overall user experience, encourage learner engagement, and realize the full potential of e-learning as a powerful educational tool, the main goal of this thesis is to identify and address the user interface (UI) challenges common in e-learning platforms. This thesis intends to contribute to continuing efforts to enhance e-learning platforms for learners globally by doing a thorough examination of current UI design methods and creating creative solutions.

To accomplish this, a User Interface will be designed and developed with will a focus on adding colorful, kid-friendly colors. By grabbing their attention and piqueing their interest in the instructional material, this strategy seeks to foster a good learning experience, especially for younger learners.

Additionally, in order to produce a user-friendly and intuitive interface, this kid-friendly UI design will not only concentrate on the use of fun colors but will also take layout, typography,

and navigational aspects into account. The entire design will put an emphasis on simplicity, clarity, and usability to make sure that young students can utilize the e-learning platform independently and develop a sense of autonomy and confidence in their academic endeavors.

1.2 Outline

The present thesis report comprises seven principal chapters, which are outlined as follows: A thorough description of the Graphical User Interface is presented in Chapter one, which also lays the groundwork for the next chapters. The theoretical and technological context are explored in Chapter two in order to clarify the key ideas and resources that drive the research. Chapter three analyzes relevant studies and provides comparisons to place the findings in the context of the body of knowledge. Chapter four provides a full description of the system and the platform design process. The implementation is covered in Chapter five, which also highlights the technologies used and goes into detail about the software components. The thesis's final results are shown in Chapter six, along with a user interface demonstration. The report is concluded with a discussion in Chapter seven, which also summarizes the results and considers their ramifications.

Chapter 2 - Background Work

The purpose of this chapter is to establish a comprehensive understanding of the thesis. For this to function, the theoretical and technological basis will be explained. More in-depth, the definition of e-learning will be analyzed, the advantages and disadvantages. Also, the User Experience definition will be described, the different types and implementations, the basis of the User Interface and principles, as well as why are they so important in software development.

2.1 E-Learning

E-learning, also known as online learning, refers to the process of gaining knowledge and skills through electronic technologies such as the internet, digital devices, and multimedia resources. The term emerged during the 1990s when the Internet started gaining popularity, and it encompasses both computer-based and web-based learning [3]. Nowadays, the term typically refers to any form of learning that can be conducted or facilitated through the Internet. As the availability of Internet access has significantly expanded, e-learning has become a popular and effective method for providing distance education solutions [2], [4].

E-learning can be classified into two categories. These are synchronous E-learning and asynchronous E-learning. Both synchronous and asynchronous e-learning have their own advantages and disadvantages, and the decision to choose between them depends on various factors, including the specific learning needs and goals of the course and its participants.

Synchronous e-learning involves real-time interaction between the instructor and learners, usually through videoconferencing, chat rooms, or webinars [4], [5]. This type of e-learning is similar to traditional classroom learning, but with the added benefit of being able to participate from anywhere. Learners and instructors can communicate and interact in real-time [4], [6], which allows for immediate feedback and engagement.

Asynchronous e-learning is a method of online learning that allows students to access course materials and complete learning activities on their own schedule, without the need for real-time interaction with instructors or peers. Students are expected to be self-directed in their learning and to engage in online discussions or other forms of communication with their instructor and classmates [4].

As e-learning continues to expand its reach and influence across various educational sectors, it is crucial to conduct a comprehensive analysis of the associated advantages and disadvantages in order to understand of its potential impact on education and identify opportunities for improvement and innovation.

Some of the advantages are:

- The potential for flexible engagement with e-learning activities in relation to other domestic responsibilities [2], [7].
- Minimal expense: The costs associated with hardware, software, and internet services are relatively low and continue to decrease over time [8].
- Students have the flexibility to work at their preferred pace and location, provided they have access to a computer [9].
- E-learning proves to be cost-efficient as it eliminates the necessity for students or learners to commute[7].

Some of the disadvantages are:

- The absence of engagement with other students or instructors might lead to feelings of seclusion and disconnection for some learners [2], [7].
- In e-learning, tests and evaluations are often overseen indirectly, which can make it challenging, if not unfeasible, to monitor or prevent dishonest behaviors like cheating [7].
- This learning strategy is that it imposes considerable responsibility upon students to determine the most suitable study approach for themselves [4].
- The rapid increase in the number of websites, with thousands of new internet sites being added daily, has made locating information increasingly challenging [8].

To summarize, e-learning continues to grow in popularity as technologies advance and educational institutions and organizations recognize its potential benefits. Following that, the User Interface definition, the importance of it and the essential principles that have been instituted. Additionally, the User Experience definition and concept will be examined, as well as the four essential components.

2.2 User Experience and User Interface

As technological advancements and evolving communication methods emerge, the significance of User Interface (UI) and User Experience (UX) is growing, leading to shifts in user anticipations and requirements [10]. A user interface (UI) represents the interactive components of a system, both software and hardware, that enable users to engage with it, input data, and utilize its contents. It was introduced in 2000s [11], and it facilitates communication between the user and the system through various commands and techniques, allowing users to perform tasks and manage information within the information system. The UI is an essential aspect of any information system, as it requires user interaction to enable the exchange of input and output [11]–[15].

The concept of user interface originates from the field of human-computer interaction (HCI) and extends beyond internet services to encompass various devices and workflows that humans interact with [16]. When interacting with the internet or digital applications, the first aspect one encounters is the user interface [16]. The design of the UI plays a crucial role in promoting a digital product, as it shapes the user experience, influences the product's success, and enhances customer satisfaction, which is vital from a business perspective [17].

Essential principles of UI design have been established based on insights from various fields of knowledge. These principles emphasize the importance of presenting information in an aesthetically pleasing and appealing manner without compromising the application's functional framework or distracting users from its primary purpose. When developing a design, it is advisable to adhere to the principle of visual hierarchy, which ranks graphical elements according to their usage priority and subordination. Moreover, the appropriate placement of visual accents, the use of contrast to draw attention to specific graphical elements, and maintaining balance in element presentation by applying the concept of visual weight are all critical considerations [17].

In the domain of digital creation, professionals differentiate between user experience (UX) and user interface (UI) design, indicating that the process of developing an interface is complex and requires a designer to possess specialized interdisciplinary knowledge [17]. User Experience (UX) encompasses the overall experience, including emotions, thoughts, reactions, and behaviors, that a user undergoes when directly or indirectly interacting with a system, product, content, or service [13], [14], [17]–[19]. UX design focuses on developing features that

enhance the user's interaction with the digital product, aiming to achieve the application's purpose and maximize user satisfaction [17].

UX, a concept related to human-computer interaction (HCI), is not only employed in software and hardware development but also extends to services, products, processes, and even society and culture [20]. The art of UX involves planning and designing a product to ensure the interaction with the completed product is as seamless and enjoyable as possible. In essence, UX captures a user's perception and response resulting from the use or anticipated use of a product, system, or service. UX emerges from a combination of the user's internal state, the designed system's characteristics, and the context in which the interaction occurs [14].

UI/UX represents an interface that allows individuals to interact with a system or application in a computing and communication setting. It can be categorized into software and hardware interfaces. Hardware interfaces consist of plugs or interface cards that connect a computer to peripheral devices, while software interfaces are exemplified by the user interface [20].

The four essential components of User Experience are Needs, Expectations, Attributes, and Capabilities. This approach addresses user needs by identifying challenges, incorporates user motivations and expectations, and takes into account societal, cultural, environmental, and informational factors as part of user attributes. Capability refers to the specific limitations, immersion, and time and space constraints for a particular user in a given environment [20].

In summary, User Interface (UI) and User Experience (UX) have gained importance due to technological advancements and evolving communication methods. UI involves the interactive components of a system, while UX focuses on overall user satisfaction. Both are crucial for promoting digital products and enhancing customer satisfaction. The distinction between UX and UI highlights the need for specialized interdisciplinary knowledge in interface development. UX design encompasses various user aspects, creating a seamless experience across services, products, processes, society, and culture.

In general, UX/UI design is a highly demand sector that calls for a variety of skills that go beyond simply those required for IT roles and include different careers like marketing, web design, and human-computer interaction. It is also stressed that UX/UI design acts as a hub where various academic fields, including basic design, engineering, marketing, and psychology, converge [17].

Chapter 3 - Similar Work

In this chapter, the focus is on the existing body of research and design that explores the role of User Experience (UX) and User Interface (UI) in the domain of e-learning. We will investigate studies and platforms that have achieved better learning outcomes for the students, through innovative ways of implementing UX and UI. In particular, the development of an adaptive UI using learning styles will be analyzed, as well as the study of User Centered design in e-learning environments. Finally, the development of STEAM, an e-learning platform using experiential learning will be reviewed.

By closely analyzing comparable research, we can glean valuable insights into the hurdles and potential advantages associated with the creation of our user interface/user experience platform, as well as benefit from the triumphs and constraints encountered in previous attempts.

3.1 Adaptive User Interface for Moodle based E-learning System using Learning Styles

An Adaptive E-Learning portal is proposed [21], that uses the Felder-Silverman Learning Style Model (FSLSM) on the Moodle Framework (figure 1). The system aims to provide a personalized learning experience by generating a customized user interface for each learner based on their learning style.

The proposed system has two modules: the first module uses web log analysis to identify the learners' learning styles, while the second module provides adaptation on the Moodle-based portal as per the identified learning styles of the learners. In addition, the proposed system provides contents as well as a customized user interface for each learner as per their learning style.

Several other works have been done using the Felder-Silverman Learning Style Model. The adaptive online module system by Hidayat A, Utomo V [22], is intended to provide students with a more tailored learning experience. The system's architecture is made up of a number of parts, including a user interface, a student model, an instruction model, a media space, a domain model, and a student model. Students fill out a questionnaire based on the Felder-Silverman

Learning Style Model (FSLSM) to start the system, and this information is then saved in the student model. The system may give adaptive learning experiences by using this data to construct a learning style profile for each student. The system achieves flexibility by providing adaptive navigation and content, choosing the best learning resources from the media library, and managing the learning flow according to the learning preferences of each student.

Another similar work is an adaptive e-learning system [23], which is built on the Moodle LMS and employs a machine learning strategy coupled with an adaptation algorithm to effectively manage expanding student data records. In comparison to complex and expensive alternative systems, the algorithm analyses records quickly to detect shared behavioral learning styles among students, enhancing performance and accuracy. The ILS questionnaire only needs to be completed once by the student, and the system will then change the student's learning styles depending on their behavior throughout the course and periodically update the student model. The study shown that dynamically adjustable LMS can enhance learning outcomes and inspire student motivation. To create efficient and useful adaptive systems, further study is necessary to understand how students learn best in e-learning environments.

The STEAM e-learning platform [24], based on Kolb's Experiential Learning Theory, aims to enhance user intentions and learning performance by blending modern interface design with a four-stage experiential learning cycle. This interdisciplinary approach places a strong emphasis on practical activities that allow students learn through reflection, experience, conceptualization, and application. The platform provides memorable experiences that learners may apply in real-life circumstances by integrating art appreciation with color application.

The study investigates how users are affected when appreciating art, concentrating on user experience and system interface usability as the cornerstones of e-learning platform design and evaluation. The platform, which was designed with a strong emphasis on usability and user experience, successfully increases learners' interest and performance across a wide range of courses. The platform is suited for a wide spectrum of learners, including learners from various departments and genders, according to the study's findings, which also point to the platform's good usability, pragmatic quality, and favorable association between ease of use and overall user experience.

In this study, the proposed architecture in figure 1, suggests that can help overcome the limitations of traditional E-Learning systems and enhance the learning experience for engineering students.
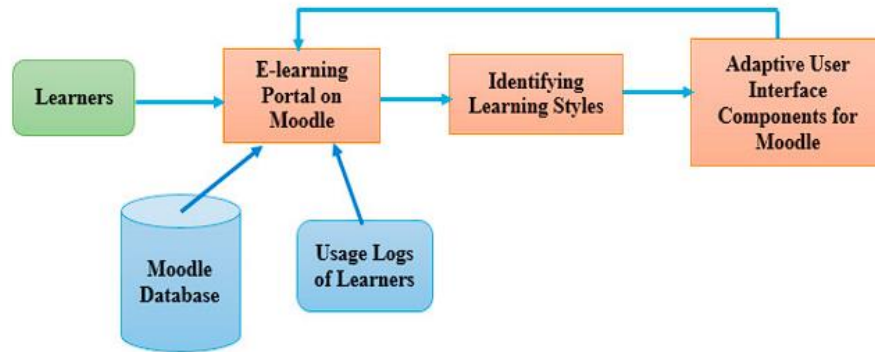

Figure 1. The Adaptive E-Learning portal architecture

This research conducted study of a pre-test and post-test experiment to evaluate the effectiveness of an adaptive e-learning system based on the Felder-Silverman Learning Style Model (FSLSM). The experiment involved randomly selecting seventy-six second-year engineering learners and dividing them into two groups: experimental and control. The learners in the experimental group were provided with an adaptive user interface based on their learning styles, while the control group received a non-adaptive interface. Both groups took a pre-test and a post-test quiz, and the results were compared using a paired t-test. The study found that identifying learning styles and recommending course contents and themes significantly increased the learners' performance.

The adaptive e-learning system adjusts the learning materials and interface based on the identified learning style of each learner. The system provides personalized course content that matches the user's learning style, using an algorithm that selects appropriate materials for each learner. Each learning style is associated with a specific learning component that facilitates understanding. To track and manage learning styles, the user profile in the database is tagged with the identified learning style of each learner, and updates are made accordingly.

3.2 User-Centered Design of e-Learning User Interfaces: A Survey of the Practices

This survey [25], discusses the importance of considering user interface (UI) quality when implementing an e-Learning system. User-centered design (UCD) is presented as the most popular practice for designing user interfaces, as it involves users in the design process to create engaging and efficient user experiences. UCD involves users in the design process to create engaging and efficient user experience. UCD is viewed as essential for ensuring e-Learning quality, as it results in better learning and information retention.

Examining seventeen research papers from 2015 to early 2020, the analysis uncovers a range of user-centered design (UCD) approaches, with the most prevalent methods being questionnaires, interviews, high-fidelity prototyping, and usability testing.

The analysis found that UCD is applied differently across projects, with methods such as questionnaires, interviews, high-fidelity prototyping, and usability testing being the most popular. Projects that involved users in multiple phases and used UCD tended to produce designs with better usability. The study also suggests that involving both learners and subject matter experts in the e-Learning system design process can lead to an excellent UI and an effective learning process.

Chapter 4 - System Analysis and Design

In the upcoming chapter, the discussion will delve into the main problem in e-learning, the critical aspects of the backend and frontend architecture, exploring in-depth the underlying components, functionalities, and features of the graphical user interface.

4.1 System Analysis

4.1.1 Problem Description

In the preceding chapters, we emphasized the significance of user interface (UI) and user experience (UX) design for digital platforms, especially in the realm of e-learning. A well-crafted UI can significantly improve the learning process and assist learners in accomplishing their objectives. When e-learning platforms have user interfaces (UI) that are difficult to understand and navigate, it can result in errors and frustration for users, particularly teenagers who frequently engage with online learning websites. The lack of design thinking implementation in many of these platforms hinders the accomplishment of learning goals and leads to an ineffective user experience [3], [18], [19].

We aim to solve this by developing a user-friendly user interface with colors suitable for children and teenagers, in order to make a better e-learning experience.

4.1.2 Backend Architecture

The Adaptive Blended Learning Platform is designed to provide a personalized learning experience to students. It is built on the 4Cs architecture, which stands for Content, Community, Collaboration, and Context, it will employ various learning methods and use learning objects to enhance the learning experience. It will also factor the individual's learning style, which will enable the platform to offer customized learning activities and resources [26].

The platform's design is centered around the learners, and it will adapt to their needs by providing them with personalized learning activities and resources. The platform will analyze the learners' learning style, progress, and performance to generate personalized worksheets. These worksheets will be based on the student's strengths and weaknesses, and they will aid in reinforcing the concepts learned.

The platform itself is split into five main modules: Learning Subjects, Learning Methods, Learning Styles, Learning Objects Management and User Profiling (figure 2).
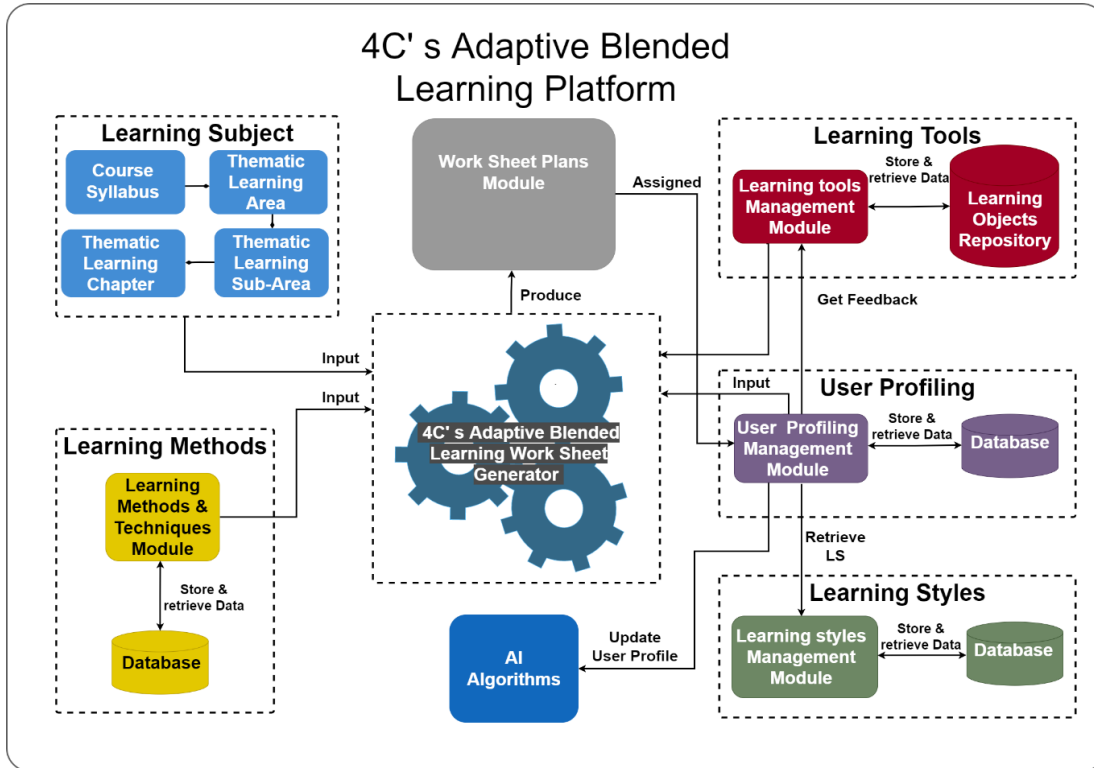

Figure 2. The Adaptive Learning Platform's Architecture

The Learning Subjects Module handles the information necessary for creating and selecting a learning subject. The learning components include details regarding the learning syllabus, thematic learning areas, sub-areas, and chapters. This information is structured hierarchically, with each learning syllabus containing thematic learning areas, and each thematic learning area containing a subset of thematic learning sub-areas, and so on.

The Learning Methods Module are a diverse set of tactics, methods, and techniques that are implemented to aid in the absorption and retention of knowledge and skills. The ultimate objective of these methods is to assist learners in interacting with and comprehending learning materials, as well as to promote favorable learning outcomes. In platform's case, the learning methods are used to filter the Learning Tools based on the teacher's preferences and the student's capabilities.

The Learning Styles Module stores the information and properties about each learning style theory like Felder and Silverman, including the dimensions and sub-dimensions of each style.

The Learning Object Management module represents methods, types and assistance for the creation of learning items. A learning object may be a video, sound, image, text, quiz or questionnaire object. Additionally, each learning object is labeled with tags, like the learning style, learning method, the type, for example if it is a learning or evaluation object.

The User Profiling Module is a crucial aspect of the platform since it holds valuable information about the user's preferences, behavior, and performance. It is the foundation of the platform's adaptivity features, allowing it to provide tailored content to the user. To ensure that the user always receives relevant worksheets, the user profile is updated each time the student completes a worksheet. Additionally, the user profile module is responsible for providing suitable material and feedback to the learner after each activity. The feedback is given through a rating system, which is designed to be quick and simple to complete. However, the student can also provide more information about their experience through an optional form. When setting up the user profile, the student will be asked to provide demographic information such as their name and grade, as well as answer a questionnaire to establish their learning style preferences.

The platform provides a classification of four roles, in addition to the Administrator role. This helps to assign specific tasks to individuals with the required expertise (figure 3). The roles are explained below:

Student: The duty of a student is to learn and finish worksheets. By using the platform, a student can access worksheets, review their contents, and edit specific fields as required. After completing a worksheet, the student should be able to offer feedback on its level of difficulty and enjoyment.

Teacher: The teacher is expected to have the capability to generate new worksheets or choose from a collection of pre-existing generic worksheets. Furthermore, the teacher has the authorization to choose and employ learning subjects and learning tools while developing a worksheet. As a result of this procedure, new learning objects can also be generated.

Content Creator: The responsibility of the content creator is to produce new learning subjects and tools for the platform. Individuals in this role study literature to identify innovative educational methods, which they incorporate into the platform as learning tools. Additionally,

they are accountable for monitoring changes in the course syllabus and updating existing learning subjects or adding new ones as needed.

Educational Expert: The task of educational experts is to review the content developed by content creators and determine its suitability for the platform. They have the responsibility of either approving or rejecting the material based on their evaluation. Additionally, educational experts are accountable for devising new learning methods and incorporating new learning styles into the platform.
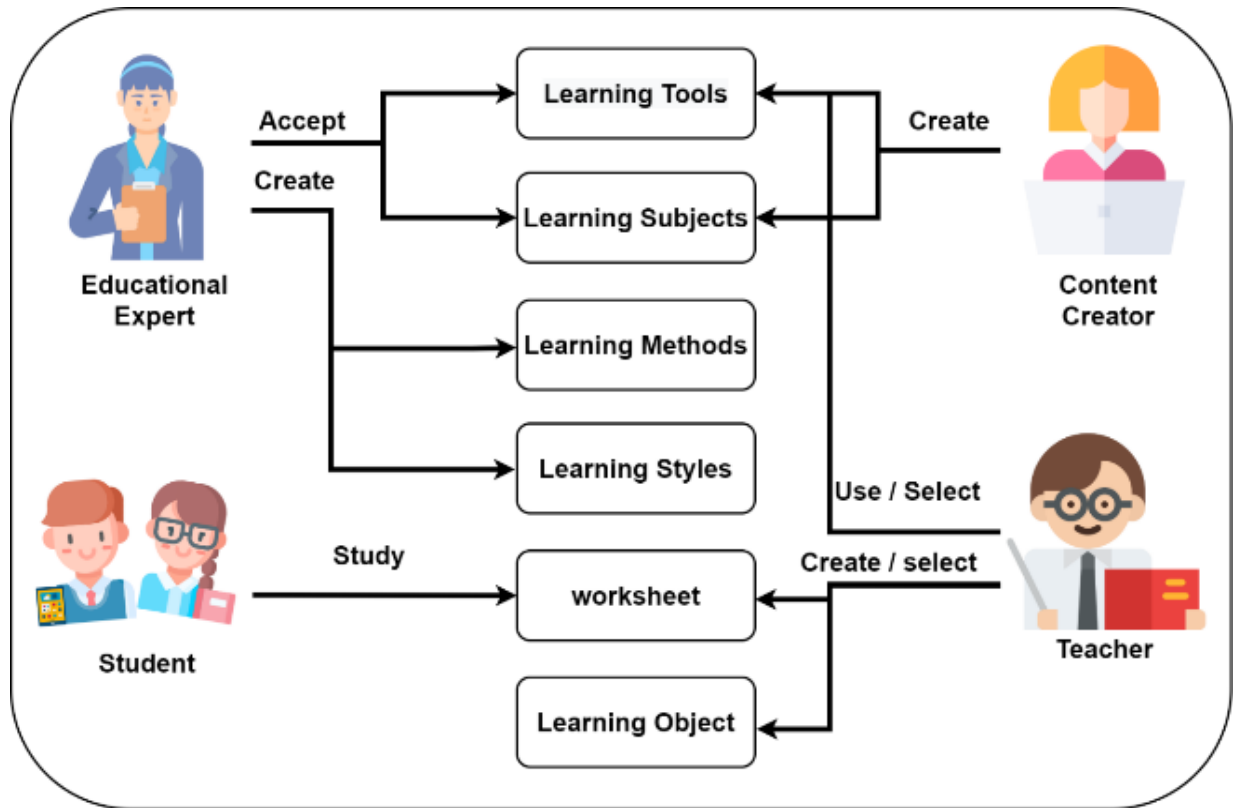
Figure 3. The Platform's Roles

4.2 System Design

4.2.1 Frontend Architecture

The platform is built with a single page architecture and is composed of three main modules: the app, router, and store state modules (figure 4). Upon initialization, the app module initializes both the router and the store state modules. The router uses nested routing to render the layout and requested pages as child routes, providing a seamless user experience. Each page in the platform contains a set of components, each of which is responsible for rendering specific content and handling user interactions. These components can modify the application state by utilizing the actions module or initiate API requests to retrieve or update data from external sources.
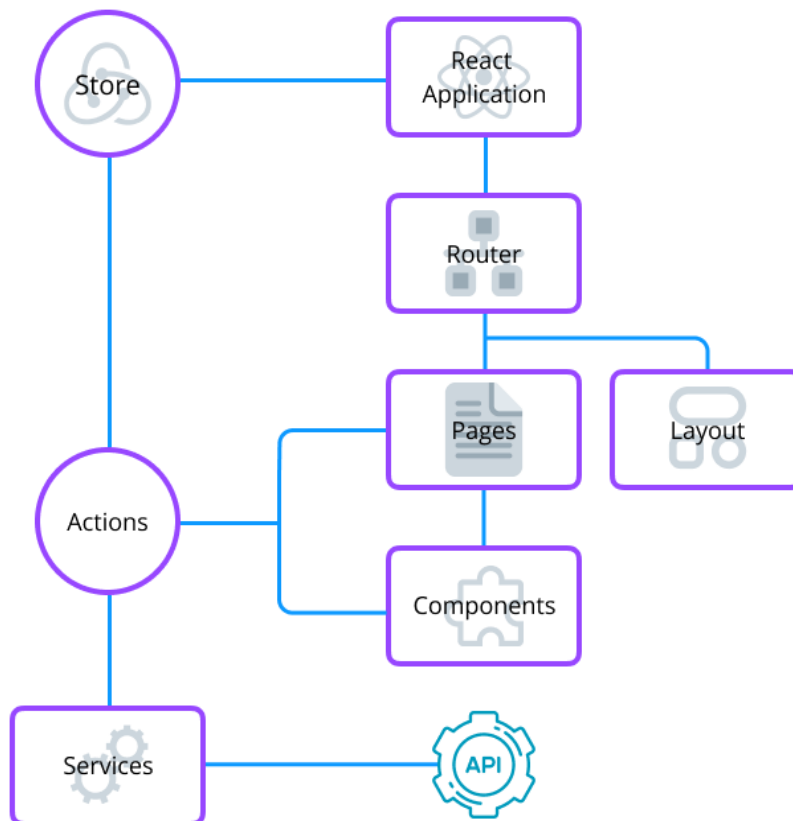
Figure 4. The Frontend Architecture

4.2.2 Components

The structure of the frontend application is intended to be straightforward and easy to expand with more features and functionalities, without sacrificing its performance or usability. This is achieved through a modular design that allows for seamless integration of new components and minimizes dependencies between them. The core of the application is the App component, which serves as the main container for all other components and manages the overall state of the application (figure 5).

The Router is created after the initialization of the App component, and it's responsible for managing the navigation and routing of the application. It enables users to move between different views or pages of the application, while maintaining the state and context of the current session. The router renders a layout component first, then it renders the requested page component within the layout, replacing only a specific section of the layout rather than the entire page.

The Page includes all the necessary API calls and components required to render the page and provide a seamless user experience. It acts as a container for the individual components that make up the page and manages the flow of data between them. Additionally, the Page component is designed to be modular and extensible, allowing for easy customization and adaptation to changing requirements.

The Service component is responsible for supplying the necessary functions that enable Pages to handle both GET and POST requests. It embodies the core logic needed to communicate with various APIs, thereby facilitating and managing the flow of data between the frontend and backend elements of the application. In essence, the Service component serves as a bridge between user interactions on the frontend and the processing, storage, and retrieval of data on the backend.

The Store Management, utilizing React Redux, houses the application's states that necessitate global access. By offering a centralized approach to govern the application's state, Store Management streamlines the development process for intricate applications and facilitates the sharing of state among various components. This central repository of states enhances consistency, maintainability, and scalability, fostering a more organized and efficient application structure.
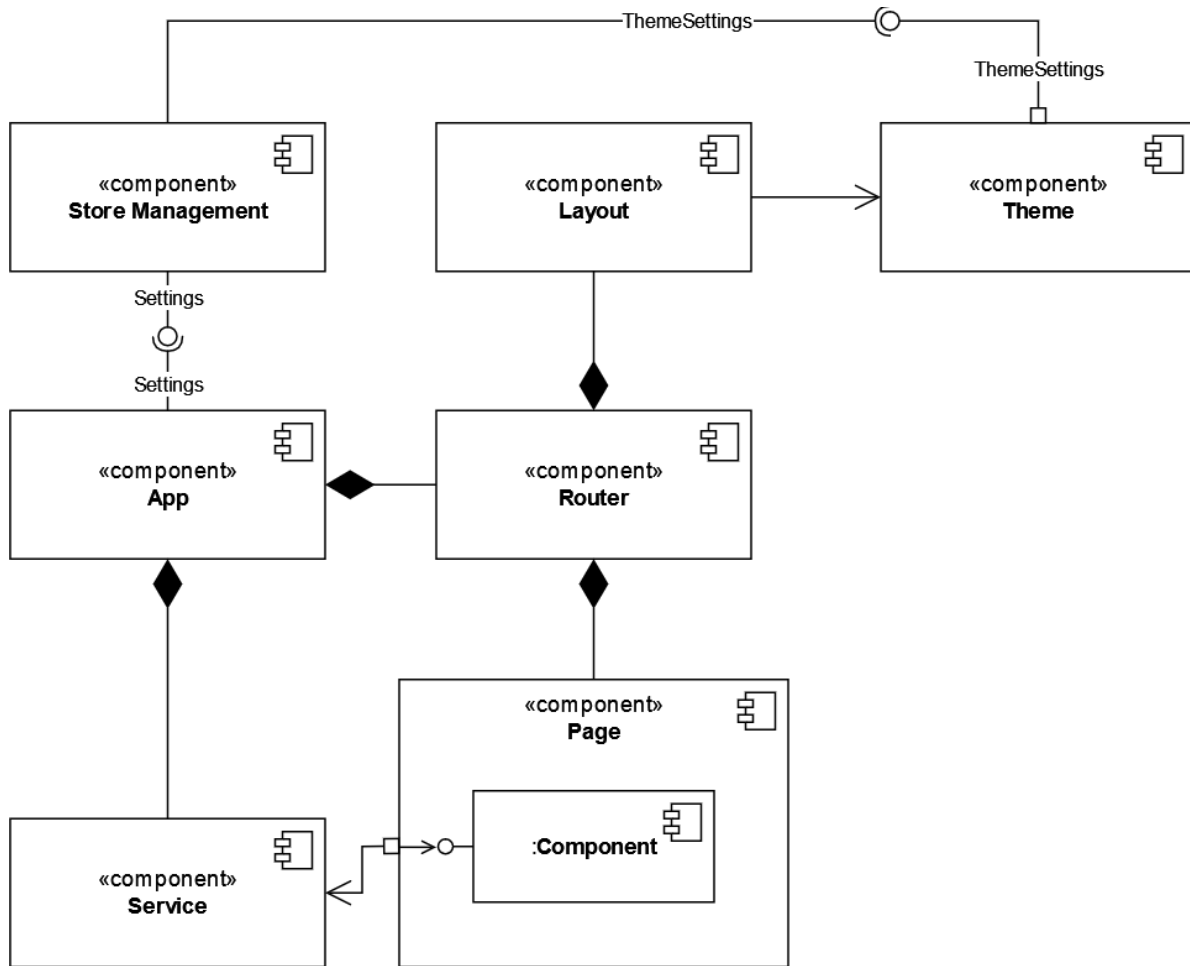
Figure 5. The Component Diagram

4.2.3 Flow Diagrams

4.2.3.1 Login and Registration

Starting by the authentication flow process, in figure 6 both the login and register authentication flow diagrams are shown. In the login flow, the user starts by filling their educational email address and the password. Then, by pressing the login button, the system will send their encrypted credentials to the back-end authentication service. If the combination of username and password is correct, the system will direct the user to the main menu page. If either the email or password are wrongly typed, the system will not direct and instead will show a warning message that either the email or password are incorrect.

In the registration flow process, the user starts by filling their educational email and password. The password is necessary to be typed twice for security reasons. After the user types the two password fields, the system will request the back-end authentication with the user's encrypted credentials. After that a confirmation email will be sent for security reasons. When the user confirms the register process by clicking the link in the confirmation email, a questionnaire will be shown in order for the system to know the user's learning style. Finally, after the user fills the questionnaire, the system will direct to the main menu page.
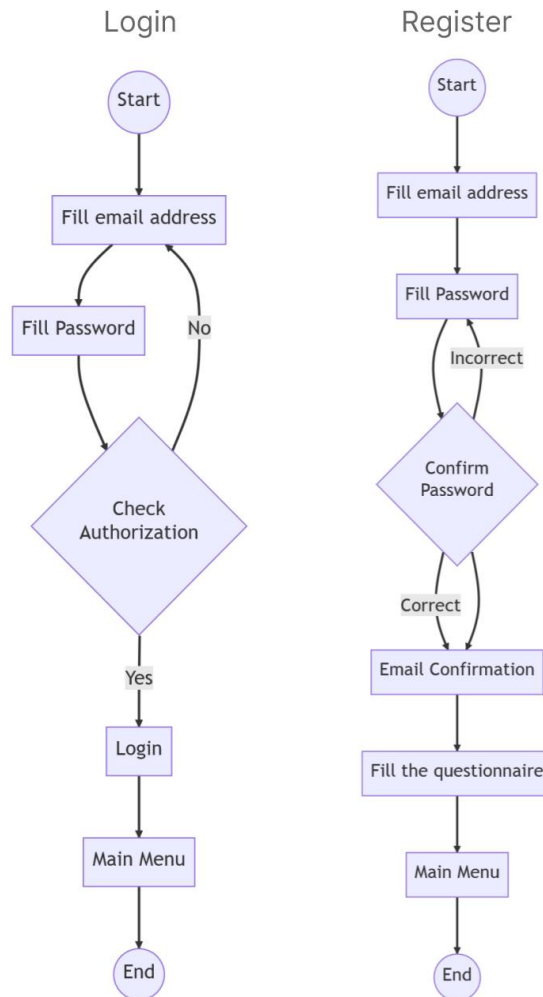
Figure 6. The Login and Register Flow

4.2.3.2 Learning Styles Components

The Learning Styles page flow starts by listing all the learning styles. In figure 7, the user's options are either viewing the details of a learning style, create new learning style or edit a learning style. If the user selects the creation of a new learning style, the creation modal will be shown with field that needs to be filled by the user. After that, the user can create the learning style. If the option is to view the details, a modal will be shown containing the details of the selected learning style. In the detail's modal, the user can either read the details, edit or delete the selected style. If the delete button is selected, the learning style will be deleted, if the edit style button is selected, the edit modal will be shown where the user can edit the fields of the learning style. By pressing the save button, the modal will be closed, and the changes will be saved.
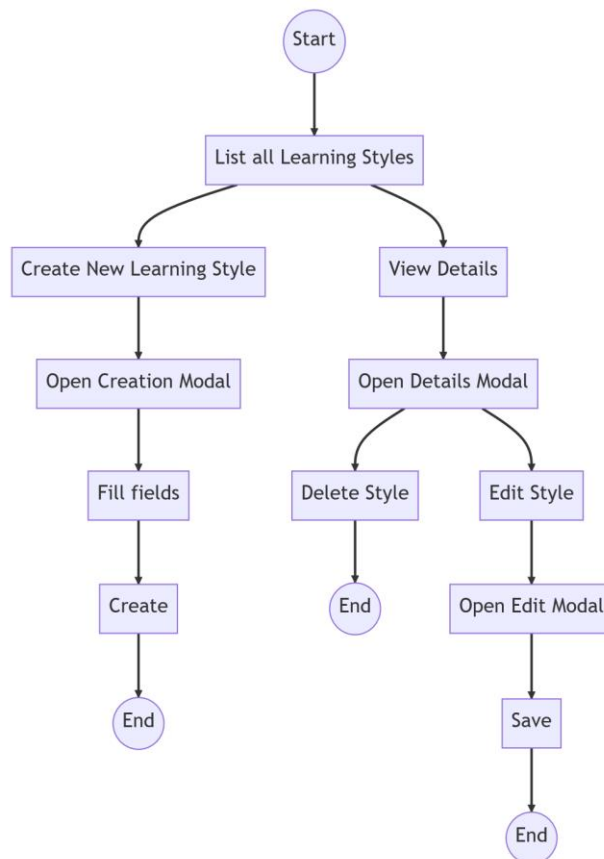
Figure 7. The Learning Styles Page Flow

19

4.2.3.3 Learning Objects Components

In the Learning Objects Page, the user's options are to open, edit and create a learning object as shown in figure 8. If the user selects to create a learning object, the system will present a page to choose the desired object to create. After that, the information is required to be filled and the object tags as well and finally the learning object will be created.

In the case of viewing a learning object, the user opens the learning object that seeks and gets directed to the view object process where all the information of the object is shown, together with the object tags. The user can also delete or edit an object via that page. Finally, the edit learning object selection or the edit object button from the view object process leads to the edit object process where the user can edit the fields of the learning object.
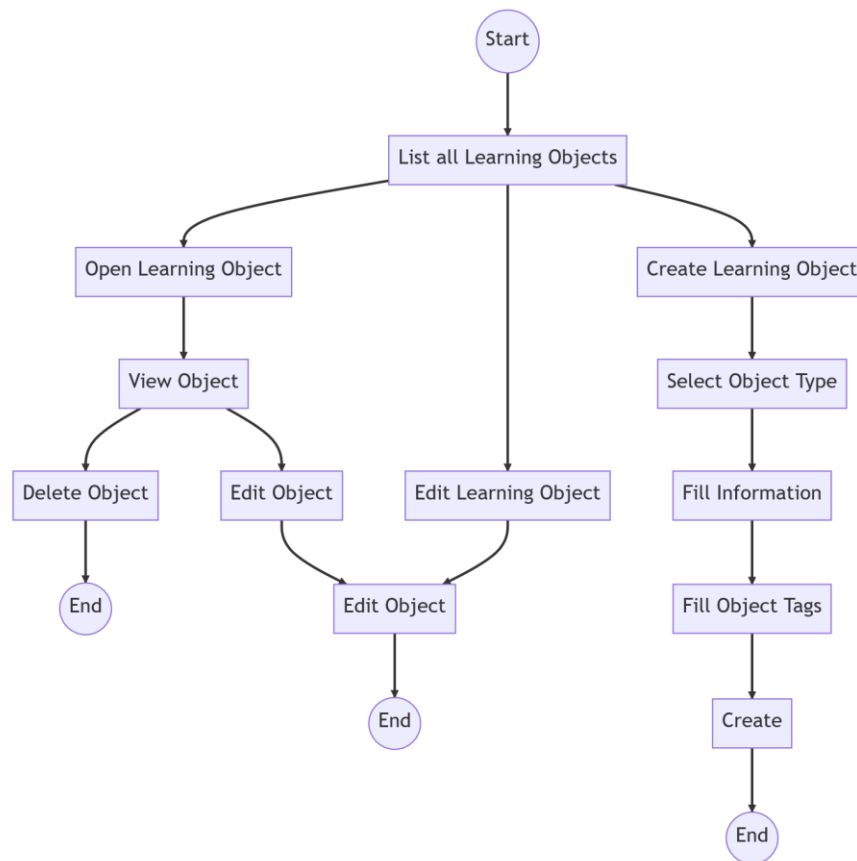
Figure 8. The Learning Objects Page Flow

4.2.3.4 The Worksheet Skeleton Creation

The figure below shows the flow diagram for creating a worksheet skeleton. The user starts by filling the basic worksheet information, such as worksheet name, language etc. After that, if the all the basic information if filled, the system proceeds the user to the next step which is the bloom level selection and 4Cs. The selection of both the bloom level and 4Cs is required, so the system will check if the user has selected them before the next step. Following that, the evaluation object selection is required, where the user must select for each bloom dimension, depending on the bloom level in the previous step. The number of evaluation objects has to be six for each dimension, two for each difficulty (None/Beginner, Intermediate, Advanced). After the selection of evaluation objects, the selection of goals for each bloom dimension is necessary, having to choose at least one goal per bloom dimension. The next step is to choose the number of learning objects for each goal. Before proceeding to the next step, the system will check if the number of learning object fulfill the selection of 4Cs. Finally, the user can review all the data in the final step before saving the worksheet skeleton.
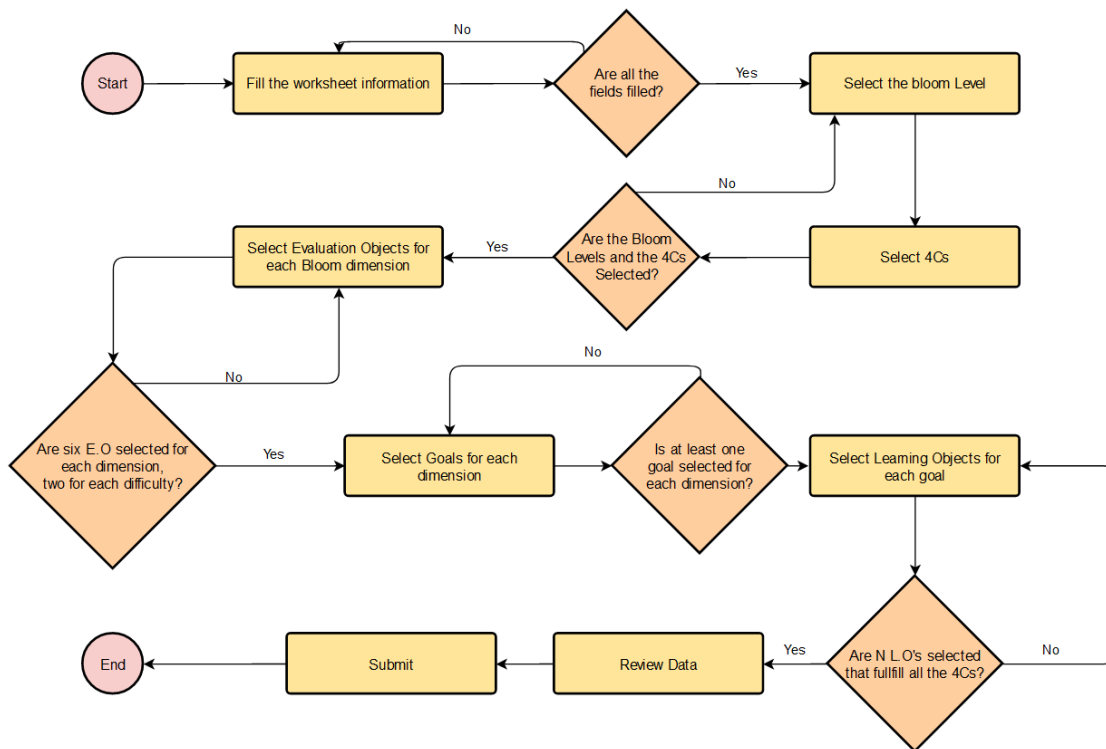
Figure 9. The Worksheet Skeleton Flow

Chapter 5 - System Implementation

The development process for the Graphical User Interface (GUI) of the adaptive learning platform prioritized maintainability and scalability, leveraging cutting-edge technologies to enhance user experience. The platform's concept revolves around delivering an intuitively designed and aesthetically pleasing environment for users, which aims to streamline the learning process and study of various worksheets.

5.1 Technologies used

5.1.1 React and Material UI

React is a JavaScript library utilized for creating user interfaces and has gained immense popularity as a front-end development tool in recent years. It was created by Meta in 2013. The architecture of React is based on components, implying that UI elements are divided into smaller, reusable pieces [27]. This makes it easier for developers to manage and update their code, as they can reuse components across different parts of their application.

React's utilizes a technique known as Virtual DOM [28]. This technique enables React to update the user interface more efficiently than conventional approaches. Instead of re-rendering the whole user interface, React solely updates the sections of the UI that have changed. Therefore, React is much faster and more efficient than other JavaScript libraries. Also, it offers a user-friendly interface that is easy to navigate, and the API is simple and intuitive making it easy for developers to comprehend and implement.

Material UI is a popular library for building user interfaces in React. It is based on Google's Material Design system, which provides a set of design guidelines and principles for creating modern and visually appealing interfaces. The library provides a wide array of customizable, pre-built UI components, such as buttons, forms, cards, modals, menus, and more. These components are responsive, meaning that they can adapt to various screen sizes and devices [29].

React and Material UI are powerful tools that can be used together to create modern, responsive, and visually appealing web applications [29]. With its component-based architecture, Virtual DOM, and support for reusable components, React provides an efficient and flexible way

to build UI components. Material UI provides a wide range of pre-built components, a powerful theming system, and support for accessibility and responsiveness, making it an excellent choice for creating the GUI's 4Cs Adaptive Educational Platform.

5.1.2 React Hook Form

React Hook Form is an open-source library that facilitates the process of creating and managing forms in React. By leveraging the capabilities of React Hooks, this library provides a more straightforward and flexible approach to form state management. The library offers a comprehensive suite of features, including input validation, conditional fields, and the ability to handle asynchronous form submissions [30]. Additionally, it boasts an intuitive and user-friendly API that developers can customize according to their specific project needs and it is particularly beneficial to developers working on high-performance projects due to its small bundle size [31]. It is also recognized for its simplicity, versatility, and efficiency, which have made it a favored choice for managing forms in React applications.

5.1.3 Vite

Vite is a cutting-edge web development tool that provides a fast and efficient development experience. It achieves this through its use of native ES modules in modern browsers, which enables fast build times and enhanced performance during development [32]. With Vite, developers can also take advantage of hot module replacement, which facilitates real-time module updates without requiring a full-page refresh. It was designed to address the limitations of existing build tools and development servers, which can be slow and inefficient, especially for large projects [33].

5.1.4 Typescript

The language that was chosen for the development of the UI's Adaptive Educational Platform is TypeScript. TypeScript a programming language that is a superset of JavaScript by adding optional static type definitions and other features [34]. It is designed to help developers write more scalable and reliable code by catching errors at compile time instead of runtime. TypeScript offers many features such as classes, interfaces, generics, and decorators, which can improve code quality and maintainability [35]. TypeScript's robust tooling and enhanced

language features contribute to the development of adaptable and efficient UIs, promoting a seamless learning experience.

5.1.5 React i18next

React i18next is a widely used internationalization library for React applications that provides an easy way to add translations and multilingual support. It supports multiple languages out of the box, and it integrates well with React and offers flexible localization options such as string translations, pluralization, and date and time formatting. Dynamic translations can be achieved by reacting to user interactions or events [36].

5.1.6 Form.io

Form.io is a platform that allows developers to create and manage dynamic web and mobile forms without having to build their own backend infrastructure. The platform provides a complete solution for creating forms, collecting data, and integrating it with other applications. Form.io's drag-and-drop form builder is a significant characteristic, enabling users to generate personalized forms with various components such as text fields, checkboxes, radio buttons, and more [37].

5.2 Software Implementation

Software Implementation plays a crucial role in the front-end development process, necessitating technical proficiency and meticulous attention to detail. This chapter will concentrate on the development of the User Interface for the Adaptive Learning Platform. Specifically, we will examine the web application's features, the organization of pages and the development of visual components. Additionally, we will delve into the analysis of complex functionalities, such as the implementation of the React Hook Form API.

5.2.1 Landing Page

The Landing Page is an introduction to the web app that contains the basic information about the platform (Figure 10). It contains four sections. The first section of the page contains the name of the Adaptive Learning Platform as well as some minimal information about it. Also, it contains an image, and more specifically a student studying worksheets, indicating that the platform generates worksheets based on user profiling.
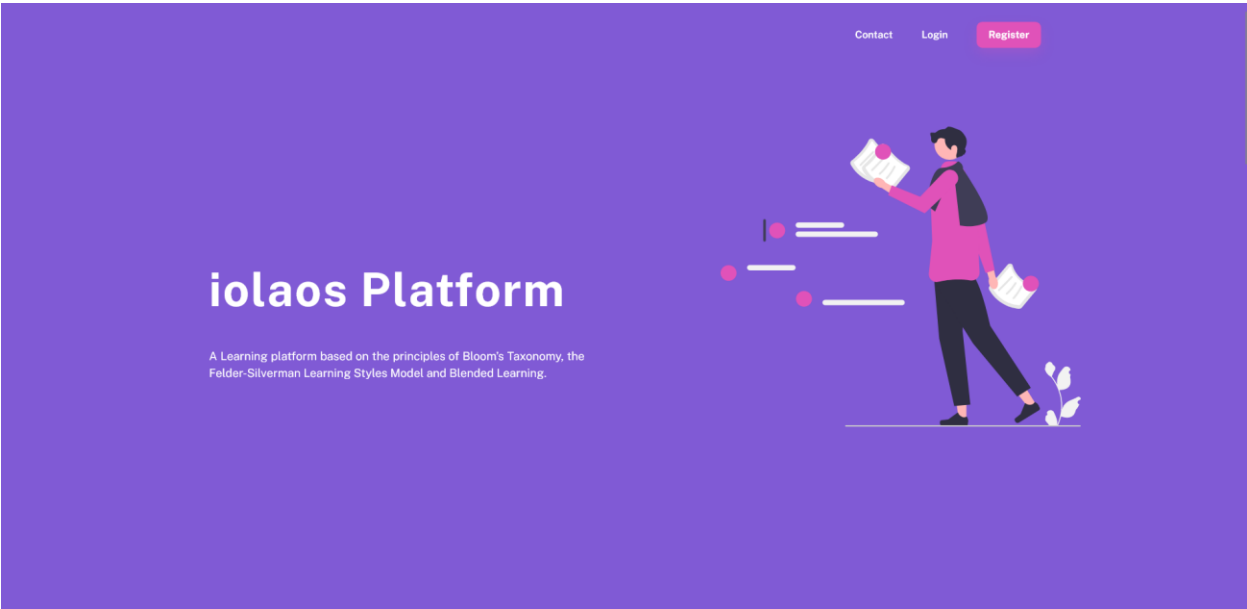
Figure 10. The Landing Page

In figure 11, the second section informs the user about the features of the learning platform and more specifically contains some information about the Quizzes, the 4C Architecture and Serious Games. Each container has an image painted with the palette's colors, which is intended to fit with the information below it.

Figure 11. The second section of the Landing Page

5.2.2 Login and Register Pages

The login and register page are made to be simple and minimal (Figure 12 & 13). On the register page the user is asked to place his email and his own password. After that the app sends confirmation email to his educational account. After the confirmation is completed, the user is prompted to answer 44 multiple-choice questions to determine the learning style.
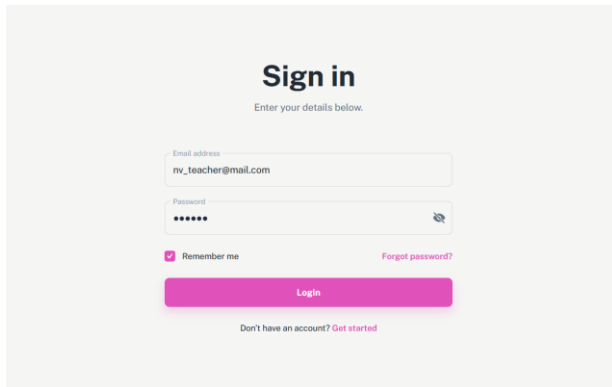
Figure 12. The Login Page
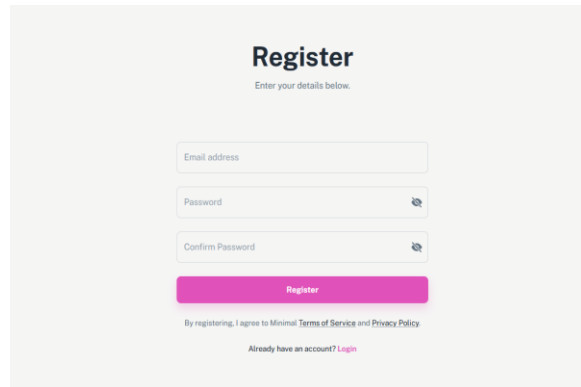
Figure 13. The Register Page

5.2.3 Main Menu

Once the user completes the questionnaire, they are redirected to the main menu page. The platform caters to various roles, including students, teachers, educational creators, and administrators. The admin role will be highlighted in this situation. The user's avatar is visible in a transparent top bar on the website. As seen in Figure 14, clicking on the avatar exposes the user's role along with "Log Out" and "Settings" choices.

Moreover, the page features six card elements, each leading to a different page. The cards' color scheme and artwork have been specifically chosen to evoke a youthful aesthetic. The visual style is reminiscent of childhood or youth culture because it uses colorful, bright hues like pastels, pinks, blues, and greens together with whimsical imagery like cartoons, animals, or playful shapes. This design strategy attempts to reach a younger audience and foster playfulness and user involvement while producing a pleasing and eye-catching user interface.
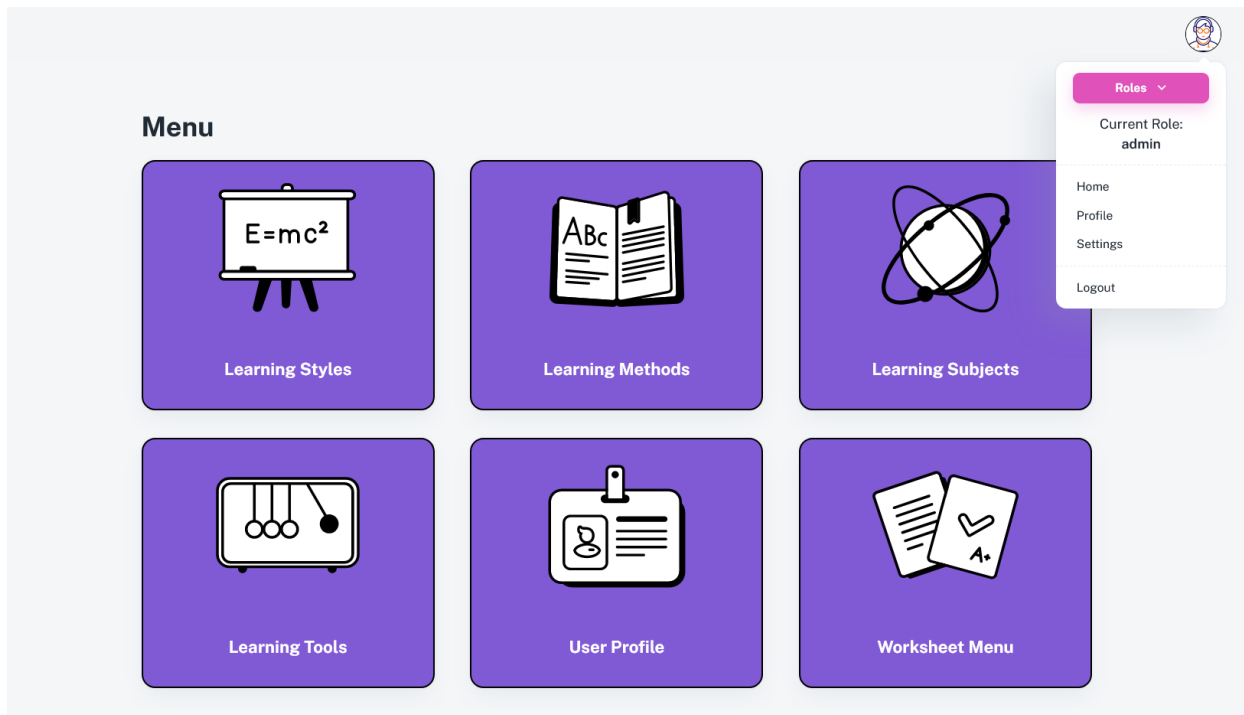
Figure 14. The main menu page with the user's control panel

5.2.4 Learning Styles

The Learning Styles page showcases an assortment of learning styles obtained from the backend, organized in an aesthetically pleasing card-like arrangement. This design choice aligns with the main menu cards' appearance, yet it intentionally omits the images on each card, putting the focus solely on the title of the related learning style.

To further support user understanding, the page includes an informative box containing essential details about the page's purpose and content. This feature assists teachers or educational content creators by setting clear expectations about what they can expect to find on the page, ensuring they can navigate and utilize the resource effectively.

The cohesive and user-friendly presentation of the various learning styles not only promotes ease of use but also encourages users to explore and engage with the content. As an added functionality, the page incorporates an "Add Learning Style" button. When users click this button, it triggers a modal window designed for the creation of a new learning style (refer to figure 15). This feature empowers users to contribute their own learning styles to the platform, fostering a collaborative and dynamic educational environment.
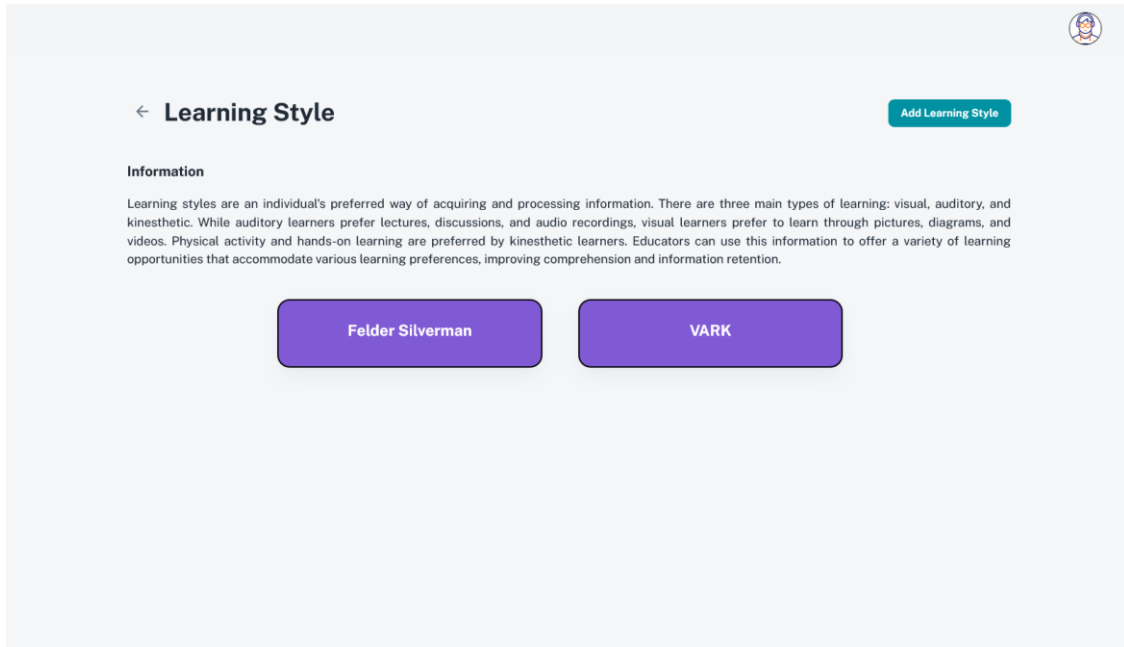
Figure 15. The learning styles page

Selecting a card opens a modal showing the details of the style the user selected. In figure 16, the modal shows the title, description, dimensions and sub-dimensions. Through the modal, if the user is authenticated, the modal provides the option to delete or edit the selected style.
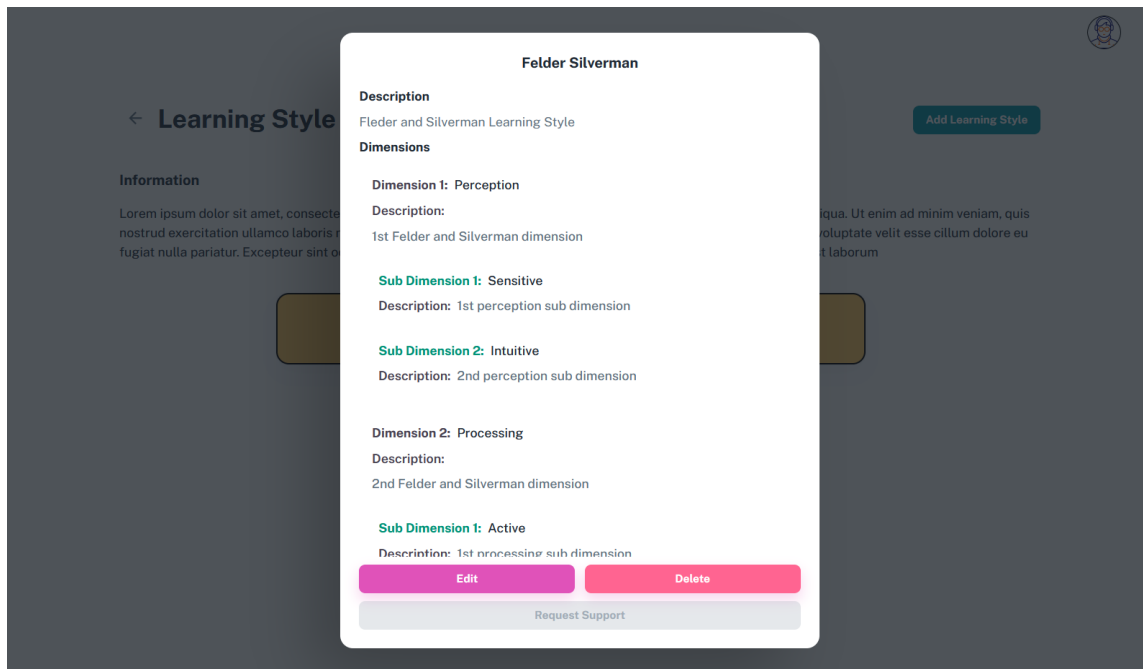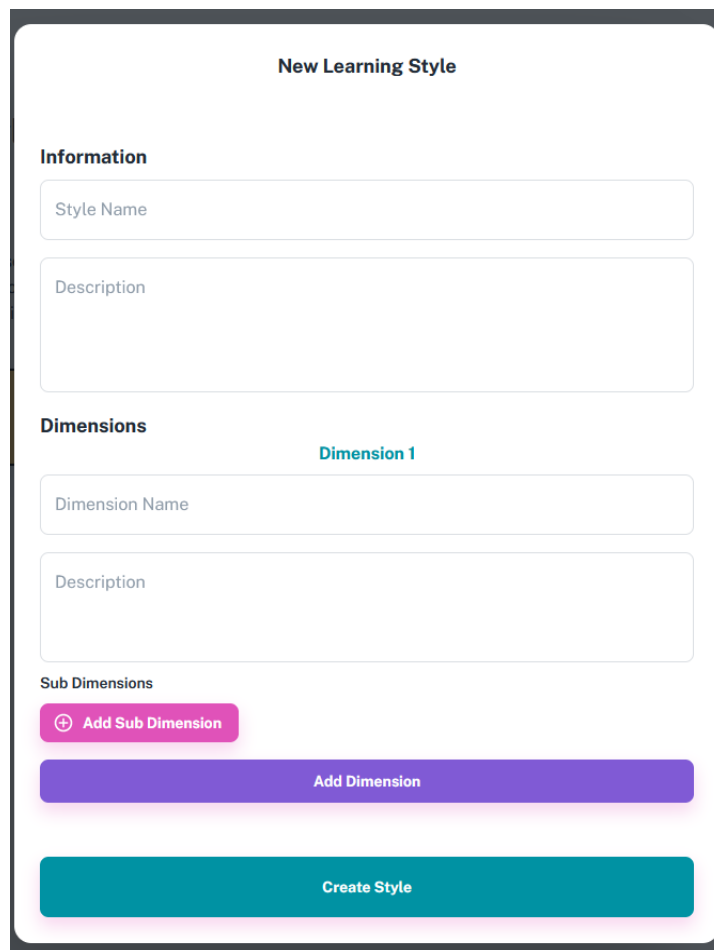

Figure 16. The Learning Style detail's modal

The application displays a modal resembling the detail's modal whenever a user chooses to add a new learning style. (Figure 17). In this instance, a form in the modal makes use of the React Hook Form API. This API makes it simple to create dynamic forms that let users enter and manage the data required for their new learning methods inside the modal.

By incorporating the React Hook Form, the application ensures a seamless and user-friendly experience, making it easy for users to contribute their own learning styles to the platform. As a result, all users benefit from a more participatory and interesting learning environment.

Figure 17. The "Add Learning Style" modal

The usability of the hook itself is very straightforward by only naming our field array and giving the control object, as shown in figure 18, which contains the other fields with their values that the hook form uses.

30

```
const { fields, append, remove } = useFieldArray({
    name: "dimensions",
    control
});
```

Figure 18. The useFieldArray hook declaration

After that, it is required to loop the dimensions array and create the field components as shown in figure 19. The dimensions initial default array size is declared in the useForm function declaration.

```
{fields.map((field, index) ⇒ {
    return (
        <Box key={field.id} >
            {/* DIMENSIONS */}
            <Typography sx={{ color: '#0092a3', fontWeight: 700 }} align='center' gutterBottom>
                {translate(`learning_styles_page.modal_dimension`)} {index + 1}
            </Typography>
            <Stack direction="row" sx={{ marginBottom: 2 }} spacing={1} >
                <TextField
                    id="outlined-basic"
                    label={translate(`learning_styles_page.name_dimension`)}
                    variant="outlined"
                    { ... register(`dimensions.${index}.name` as const, {
                        required: true
                    })}
                    defaultValue={field.name}
                    fullWidth />
            </Stack>
            <Stack>
                <TextField
                    id="outlined-multiline-static"
                    label={translate(`learning_styles_page.name_dim_descr`)}
                    multiline
                    rows={3}
                    { ... register(`dimensions.${index}.description` as const, {
                        required: true
                    })}
                    defaultValue={field.description}
                />
            </Stack>
            <SubDimensionFields nestIndex={index} { ... { control, register }} />
            <Stack direction='row' spacing={1} mb={2} >
                {getValues('dimensions').length > 1 ?
                    <Button
                        onClick={() ⇒ remove(index)}
                        color="error"
                        variant='contained'
                        fullWidth>
                        {translate(`learning_styles_page.modal_delete_dimension`)}
                    </Button> :
                    <Button
                        onClick={() ⇒ append({ name: '', description: '', sub_dimensions: [] })}
                        color="secondary"
                        variant='contained'
                        fullWidth
                    >
                        {translate(`learning_styles_page.modal_add_dimension`)}
                    </Button>
                }
                {/* To add the "add dimension button" under the last added dimension */}
                {getValues('dimensions').length > 1
                    &&
                    getValues('dimensions')[index + 1] === getValues('dimensions')[-1] ?
                    <Button
                        onClick={() ⇒ append({ name: '', description: '', sub_dimensions: [] })}
                        color="secondary"
                        variant='contained'
                        fullWidth
                    >
                        {translate(`learning_styles_page.modal_add_dimension`)}
                    </Button> : <></>
                }
            </Stack>
        </Box>
    )
})}
```

Figure 19. useFieldArray hook in learning styles creation

For the sub-dimension fields, the database model requires to have an array inside the dimensional array. In order to implement this, another sub dimension array was created inside the useFieldArray hook called "sub_dimensions". This array is created on every index of the dimensions array. After that the same method was used to create the sub-dimension fields as the dimension fields.

If the user selected to edit a learning style, an edit modal will be opened, showing all field prefilled with the information from the detail's modal. The form is structured the same way as the new modal form, with the only difference being the default values in the useForm function (Figure 20).

```
const { control, register, handleSubmit, getValues } = useForm<FormValues>({
    defaultValues: {
        name: dataModal.name,
        description: dataModal.description,
        dimensions: dataModal.dimensions
    },
    mode: "onBlur"
});
```

Figure 20. The default values in edit modal

5.2.5 Learning Methods

The Learning Methods page follows the same structure as the Learning Styles page. The cards are designed with the same color and font as the learning styles cards, as well as the details, edit and the new modal. The page also contains filtering, and more specifically the user can use sort alphabetically the learning methods and can search by title (Figure 21). The fields in creation and edit modals are different from the styles' modals, since the database model is different from the learning styles model. As a result of this, details modal shows the name, description, levels and sources. At least one level is necessary: name, description and with optional notes and examples.

The new and edit method modals follow the same programming structure, using the React Hook Form for manipulation of fields. The useFieldArray hook is used on levels field, as well as on notes, steps and examples. Since a level might contain multiple notes, steps and examples, the useFieldArray hook was used on each one respectively to enhance the dynamic

form. The edit modal shares the same pattern as the new modal, with the only variation between them being the default values in the fields.

Figure 21. The Learning Methods filtering box

5.2.6 Learning Subjects

This Learning Subjects page is divided into three sections: subjects, schools and courses. Each section is shown with tabs, with each one sharing the same cards and modals like the other pages (Learning Styles, Learning Methods).

More specifically, the contents of each tab are manifested with pagination and filtering. The introduction of pagination was necessary, with the aim to make a better experience for the users and help find the information they need easily. The filtering in subjects' and schools' tab is presented with and accordion (Figure 22). The filtering in the courses' tab is presented with a container that has a search box. The create, edit and delete operations are designed with the same pattern as the other sections and pages, with the content being, adapted to the subject's backend.
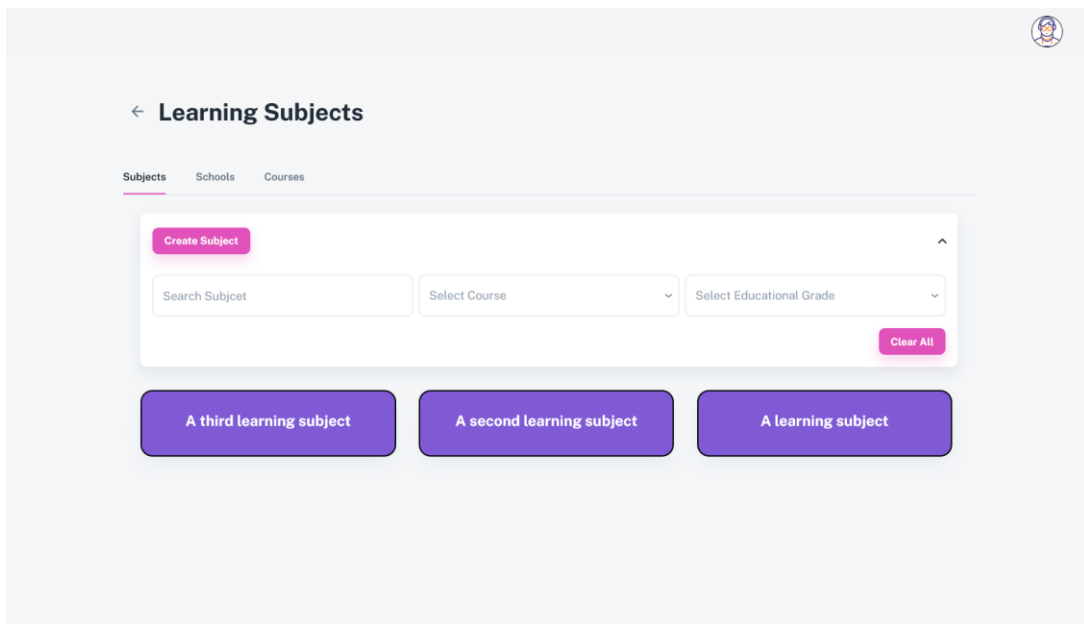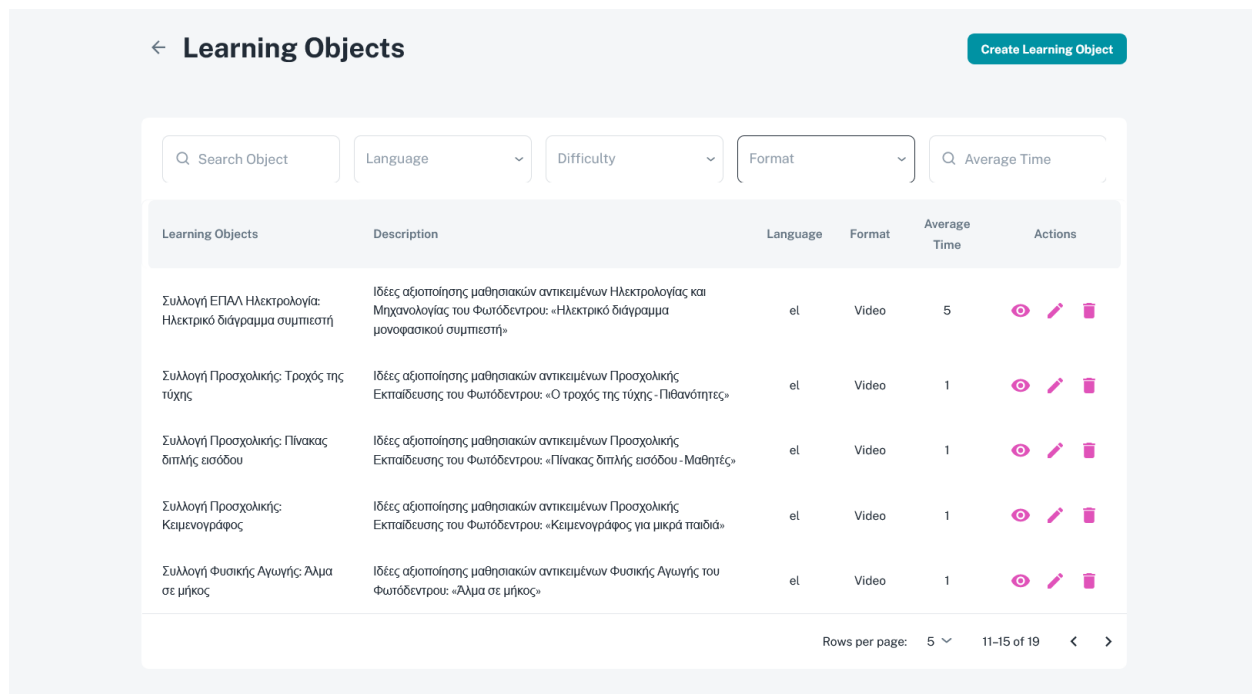
Figure 22. The "Learning Subjects" page with the filtering accordion

5.2.7 Learning Objects

The Learning Objects page is distinctive from the other pages. More specifically, when the authenticated user selects this page, a list of learning objects is shown in table form. The table shows the details of each learning object, with each row additionally containing a view and edit object button (Figure 23). The table also contains filtering, like selecting the difficulty of the object, language, format and the average time taken.

Figure 23. The Learning Objects page

The presentation of the learning object is displayed on a new page that appears when the view objects button in the action column is clicked. Based on the unique identification (id) that was provided as a parameter in the URL, this is. The presentation page displays all relevant details about the learning object, including any files that are attached and any associated tags.

The context of the object is immediately understood by educators because to these tags' insightful information. The file associated with the learning object may be a video, audio, image, or text item that was downloaded from the internet or was uploaded directly from the user's computer. (see Figure 24). The learning object could occasionally include a quiz or questionnaire.

Learning Object Presentation

Title
Συλλογή Φυσικής Αγωγής: Άλμα σε μήκος

Description
Ιδέες αξιοποίησης μαθησιακών αντικειμένων Φυσικής Αγωγής του Φωτόδεντρου: «Άλμα σε μήκος»

Language	**Format**	**Time (minutes)**	**Visibility**
El	Video	1	Public

Type	**Structure**	**Learning Method**
Open	Atomic	Inclass

Bloom Dimensions
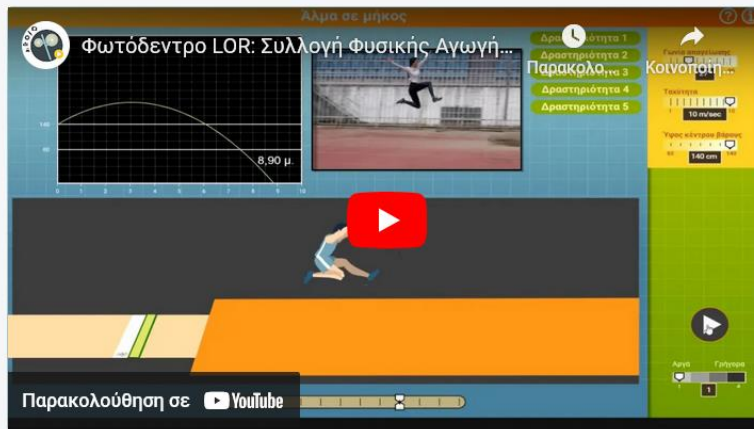evaluating, analyzing

4Cs
Communication, Collaboration

Educational Grades
Προνήπια ή Μικρά Νήπια, Μεγάλα νήπια

Learning Subjects
A third learning subject, A second learning subject

Learning Styles
Felder Silverman

Object Types
Evaluation, learning

Video Preview ⓘ

Edit   Delete

Figure 24. The Learning Object Presentation Page

35

The process of creating a learning object is designed to be simple and user-friendly. By clicking the "Create Learning Object" button, users can effortlessly choose the type of object they want to create. For instance, as demonstrated in Figure 25, selecting the "Create Video Object" option directs users to the "Create Video Object" page.
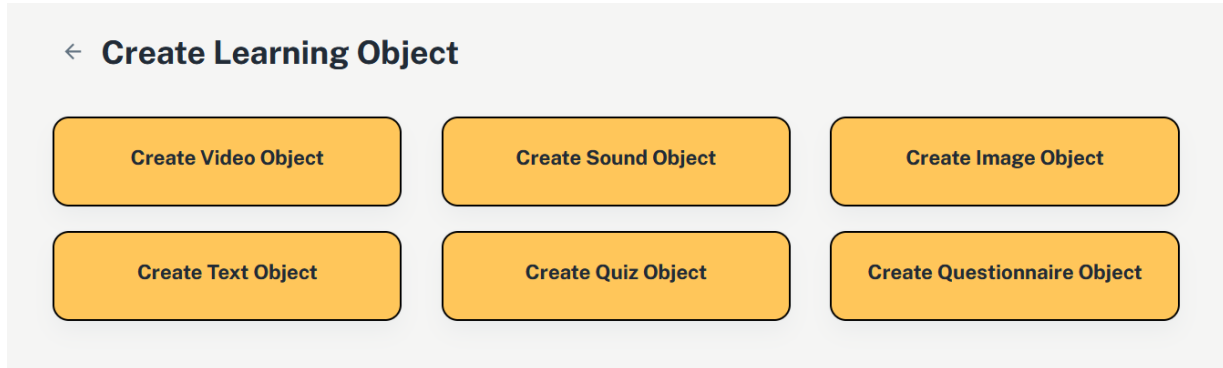

← Create Learning Object

Create Video Object	Create Sound Object	Create Image Object
Create Text Object	Create Quiz Object	Create Questionnaire Object

Figure 25. The Learning Objects Creation Menu

This page contains a form with the all the object details to be filled. All the fields are required, together with a video placed either from a URL or uploaded from the desktop (Figure 26). In order to implement the feature of showing and playing the videos, the library react-player was chosen. React player is lightweight and efficient, easy to integrate and designed to work well on various screen sizes, ensuring a consistent user experience across different platforms.

For the ability to have a player for audio, the react-audio-player was implemented as it is very easy and straightforward to use. In order to implement the feature of viewing and placing PDF files to the Learning Object, React Iframe was utilized, which is a convenient Typescript– enabled wrapper around the native HTML iframe tag.

The React Iframe component is effective and practical for adding iframes to React applications, enabling the smooth integration of external content. React Iframe, a flexible wrapper over the standard HTML iframe tag, makes embedding content into current web pages simple by enabling developers to show a range of material, including web pages, documents, and video. Developers find it appealing because of its ease of installation and TypeScript compatibility. React Iframe supports responsive layouts through CSS and allows adjustment of iframe characteristics and features including width, height, and source URL. This guarantees a

same user experience on a range of gadgets and screen sizes. React Iframe increases versatility and user experience by enabling the seamless integration of external information.

Figure 26. The creation of video object page

For the Quiz and Questionnaire object, a form builder playground had to implemented in order to provide to the teacher the ability to feature different components for the quiz or questionnaire object (Figure 27). These may be a text box or a multiple select question field. For this use case, the formio library was selected, as it has features that make it accessible and useful for non-technical users who need to create and manage forms without the need for programming knowledge or skills. Also, formio features intuitive user interface, with user-friendly menus and the option to drag-and-drop components into the form.

Figure 27. The Form Builder Playground with the Formio library

After the info is filled, the user can proceed to the next step, which is the object tagging (Figure 28). The object tagging is a component used on all the different learning objects, since all objects have the same tagging. The component includes multiple select fields, which in some of them the user can select multiple values. There values are represented like chips to indicate that can be multiple selected and can be removed by clicking again on the menu item.

Figure 28. The Object Tagging component

The creation page is also used as edit page, since the difference is that the fields are prefilled, together with their tags and the title of the page is named Edit Object. In figure 29, depending on the state of the page, the "Save" button can either work as POST or PATCH request. If it's a POST request, the createLO is called in order to create the new object. If it's a PATCH, the updateLO function is called in order to edit the object.

```
try {
    if (!mode || mode === "CREATE") {
        const createdLO = (await createLO(finalLO)).data;
        alert('Learning Object Created');
        navigation(0)
    } else {
        if (l_object) {
            const updatedLO = (await updateLO(l_object.id, finalLO)).data;
            alert('Learning Object Update Completed');
            navigation(LO_MAIN_PATH)
        }
    }
} catch (error) {
    alert('Failed to create try again');
}
```

Figure 29. The state of the submit at the object creation page

39

5.2.8 User Profile

In comparison to other pages, the User Profile Page demonstrates a more minimalistic approach. The page presents with an overview of the user's profile with the current and past occupation split in tabs. In particular, the Overview tab shows his avatar in larger dimension with the information about the user, such as username, email and phone number, as shown in figure 30, while the Occupation tab shows detailed info about his current and past occupation.
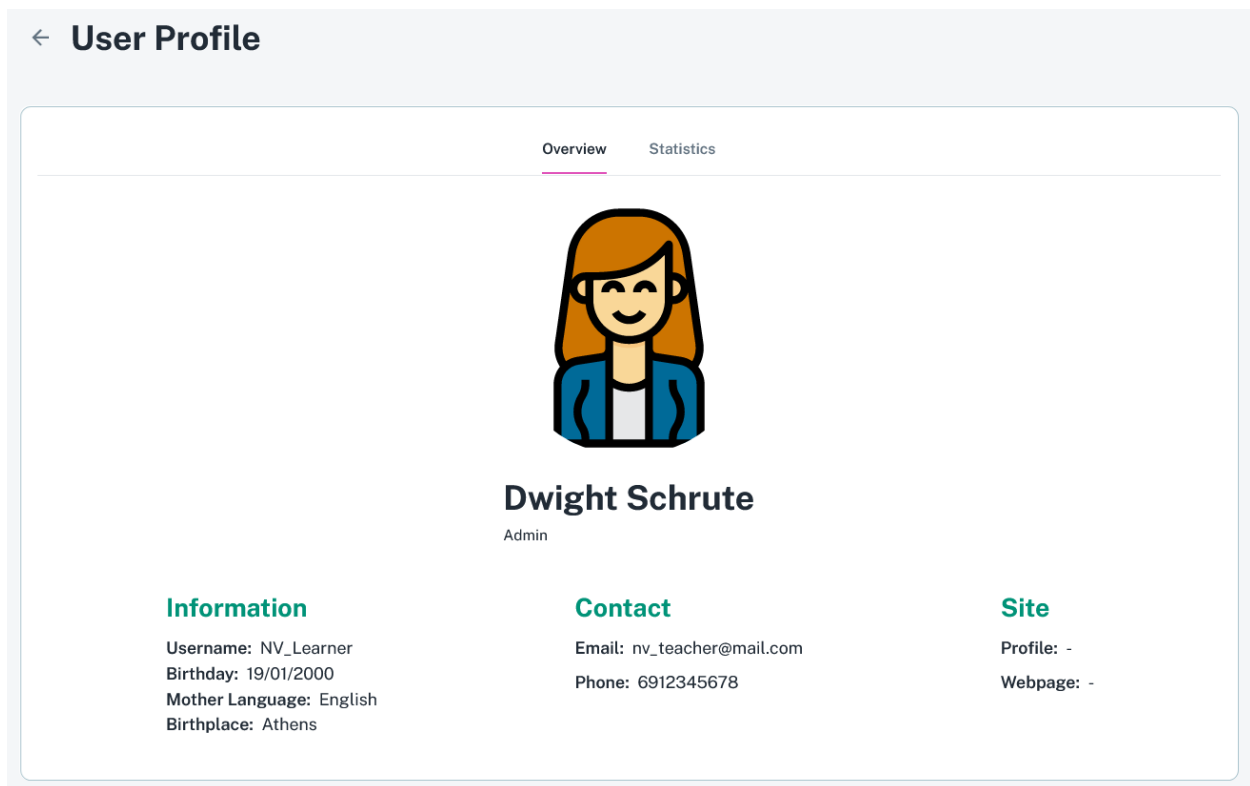
Figure 30. The User Profile page

5.2.9 Worksheet Skeleton Creation

This page is visible and accessible only to the authenticated users. These users may be the administrator, the content-creator or the teacher. The creation of the Worksheet Skeleton is structured as a stepper, with a step-by-step form using react hook form. If a student tries to access the page, a redirection will occur to the login page.

The first step is to fill the worksheet information, such as the worksheet name, description of the worksheet, the language and the educational level, shown in figure 31. All

these fields are required, meaning that the user won't be allowed to proceed to the next step if one of the fields have been left empty. Instead, the hook form will highlight the fields that have been left empty.

The second step is to select Bloom level and which 4C the user desires. Both fields are required, and for the 4C select field at least one 4C is required to select. The bloom level fields are three: "All", "Low", "High". Depending on the selection, the bloom dimension will be selected with that dimension level type. The 4C select field is multiple and highlighted with Material chips to indicate which 4Cs the user selected.
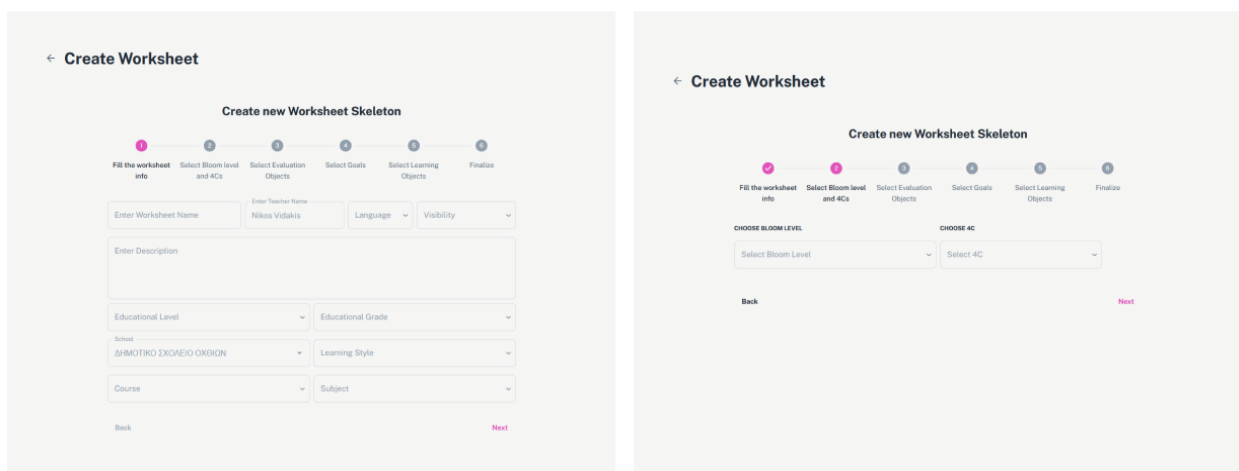
Figure 31. The step one & two in the Worksheet Skeleton Page

The third step is to select evaluation objects for each Bloom's taxonomy dimension. The evaluation objects are learning object with having evaluation as object type. Therefore, the implementation of the step is incredibly easy, as the component require minimal changes.

Each evaluation object should align with the chosen Bloom's dimension, and the user must select at least six objects, comprising of two objects with high-level difficulty, two with moderate-level difficulty, and two with low-level or introductory difficulty (figure 32). Every row click triggers a code snippet validation, which is used. Each time a new version of the page is rendered, the validation determines if the conditions are met for each selected row by adding it to a React state. Users cannot move on to the next phase until all conditions are satisfied. This validation process guarantees that the evaluation objects selected meet the essential standards, fostering an organized and productive worksheet skeleton making for all users.

41

```
import { EvaluationObjectState } from "../type/EvaluationObjectState";

export const validateEvalState = (state: { [key: string]: EvaluationObjectState[] }): boolean => {
    // to avoid returning in the useEffect and enabling with the next button
    if (Object.keys(state).length > 0) {
        for (const dimension in state) {
            const objects = state[dimension];
            if (objects.length !== 6) { // check if dimension array has exactly 6 objects
                return false;
            }
            let noneCount = 0;
            let intermediateCount = 0;
            let advancedCount = 0;

            for (const obj of objects) { // counting the difficulties in a dimension array
                switch (obj.difficulty) {
                    case "NONE":
                        noneCount++;
                        break;
                    case "INTERMEDIATE":
                        intermediateCount++;
                        break;
                    case "BEGINNER":
                        noneCount++; // none / beginner are the same
                        break;
                    case "ADVANCED":
                        advancedCount++;
                        break;
                    default:
                        return false;
                }
            }
            // check if dimension array has exactly 2 & 2 & 2
            if (noneCount !== 2 || intermediateCount !== 2 || advancedCount !== 2) {
                return false;
            }
        }
        return true;
    }
    return false;
};
```

Figure 32. The validation logic for the evaluation objects

The fourth step is to select the goals for each Bloom dimension. In figure 33, the number of tabs depend on the Bloom level the user selected in the previous step, in which every tab is named after each bloom dimension that got selected from the second step and contains all the goals designed for that dimension.

The requirements are that for each dimension, at least one goal must be chosen in order to move forward. Above each goal category, their category tags are shown in order to guide the teacher or educational creator on what to choose.
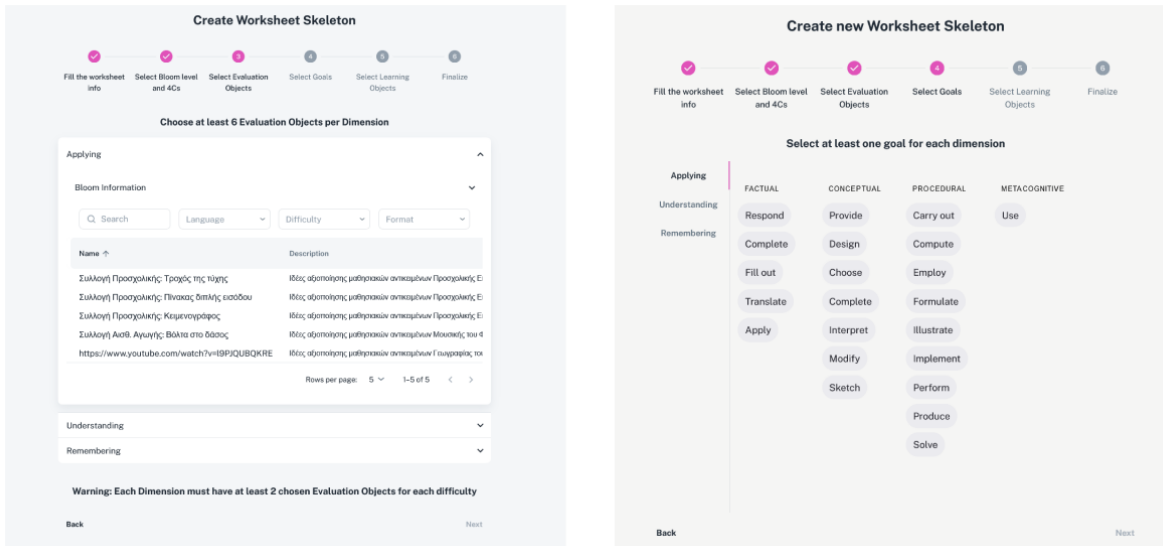
42

Figure 33. The step three & four in the Worksheet Skeleton Page

The fifth step is to select for each goal at least one learning object. The learning objects themselves that are presented, are compatible with each chosen bloom dimension from the previous step and compatible with the 4Cs. The requirements for this step are for the user to choose at least one learning object per goal that satisfies the 4Cs selections. The validation executes on every table row click (Figure 34).

```typescript
import WSInfoData from "src/config/types/createWorksheet/worksheetInfoStep/WSInfoData";
import { LearningTablesState } from "../types/LOPageStates";

const validateObjectState = (state: LearningTablesState, data: WSInfoData, chosen4Cs: string[]): boolean => {

    // Find the 4cs the user chose from bloom selection step
    const fourCsNames = data.fourCsData.filter(element => chosen4Cs.includes(element.value))
    const fourCIds = fourCsNames.map(element => element.id);

    // to avoid returning in the useEffect and enabling with the next button
    if (Object.keys(state).length > 0) {
        for (const tableNames in state) {
            const objects = state[tableNames];

            let object4CsIds: string[] = [];

            for (const object of objects) {
                for (const fourC of object.four_cs) {
                    object4CsIds.push(fourC.four_cs_id)
                }
            }
            if (!fourCIds.every(elem => object4CsIds.includes(elem))) {
                return false;
            }
        }
        return true;
    }
    return false;
};

export default validateObjectState;
```

Figure 34. The validation logic for the learning objects

To provide a better user experience, a loading bar exists to indicate the number of goals the user has completed. The learning objects are presented with a table and filtering, so the selection can be faster. The selection of the learning object can be achieved by clicking on a table row. If a learning object is selected, it will be indicated by staying in selected. Bellow the table a button exists to help the user advance to the next goal (Figure 35).

Figure 35. The fifth step in the Worksheet Skeleton Page

The final step is to review the chosen data from the previous steps. The first 2 steps are represented in accordions, in which the data is put into disabled text fields (Figure 36). The

evaluation and learning object respectively are shown or hidden depending on the view state the page currently has. The initial view state is to show the evaluation objects table. When the user clicks the finish button, the request body is sent to the server and a new worksheet skeleton is created.
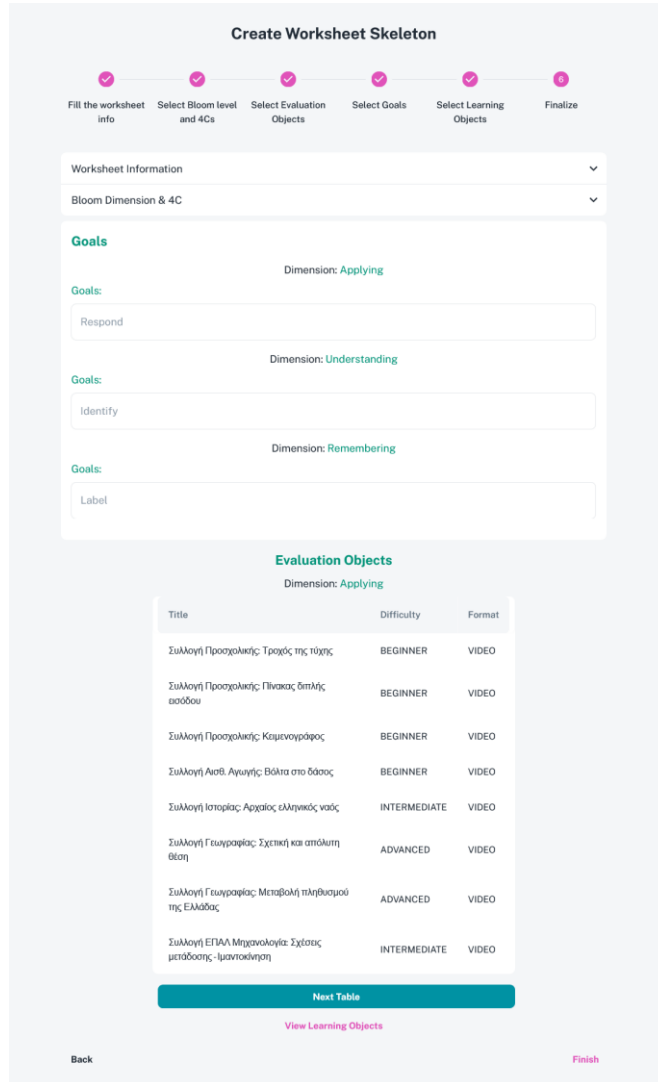

Figure 36. The final step in the Worksheet Skeleton Page

5.2.10 Dark mode and Localization

The GUI support dark mode, in which is a color scheme that displays light-colored text and icons on a dark background, as opposed to the traditional light mode that displays dark text and icons on a light background. This can reduce the eye strain if the user spends a lot of time in

45

the platform and improves visibility in low-light conditions. The Material UI provides components with dark mode functionalities, as a result the implementation was simple. In figure 37, a redux-toolkit settings slice had to be implemented in order to notify the system that the user requested for dark mode and the switch happens when the toggleMode function in triggered.

```
const initialState: SettingsState = {
    show: false,
    themeMode: defaultSettings.themeMode,
    themeContrast: defaultSettings.themeContrast,
    themeColorPresets: defaultSettings.themeColorPresets,
};

const settingsSlice = createSlice({
    name: 'settings',
    initialState,
    reducers: {
        toggleDrawer: (state) => {
            state.show = !state.show;
        },
        toggleMode: (state) => {
            state.themeMode = state.themeMode === 'light' ? 'dark' : 'light';
        },
        changeMode: (state, action: PayloadAction<ThemeMode>) => {
            state.themeMode = action.payload;
        },
        toggleContrast: (state) => {
            state.themeContrast = state.themeContrast === 'default' ? 'bold' : 'default';
        },
        setThemeContrast: (state, action: PayloadAction<ThemeContrast>) => {
            state.themeContrast = action.payload;
        },
        setThemeColorPresets: (state, action: PayloadAction<ThemeColorPresets>) => {
            state.themeColorPresets = action.payload;
        },
        resetSettings: () => initialState
    },
});
```

Figure 37. The Settings' redux-toolkit slice

The platform offers support for Greek localization, allowing users to easily switch between languages and access content in their preferred language. This feature ensures that the platform can effectively cater to Greek-speaking users, providing them with a more personalized and user-friendly experience. The localization is provided by i18n (figure 38), detects the default

language the system has and lets the user switch fast and easily languages through the click of a button without refreshing the page.

```
i18n
  .use(LanguageDetector)
  .use(initReactI18next)
  .init({
    resources: {
      en: { translations: enLocales },
      gr: { translations: grLocales }
    },
    lng: localStorage.getItem('i18nextLng') || defaultLang.value,
    fallbackLng: defaultLang.value,
    debug: false,
    ns: ['translations'],
    defaultNS: 'translations',
    interpolation: {
      escapeValue: false,
    },
  });

export default i18n;
```

Figure 38. The i18n localization function

Chapter 6 - Implementation Results / Demo

This chapter will present the outcomes of the practical aspect of this thesis, which focuses on the development and implementation of a graphical user interface (GUI) for the adaptive learning platform. The user experience and feedback of several pages and components will be presented, including the creating a learning method, the addition of a new worksheet skeleton and a new learning object. Additionally, the settings panel will be presented, along with the localization of the user interface and the dark mode.

6.1 Creating A New Learning Method

When the user clicks the "Add Learning Method" button, a new modal shows up with the fields required to fill in order to submit the new method. All level field are dynamic, meaning

that the user can add one or more levels if it is necessary, along with the note, steps, examples and sources (Figure 39).
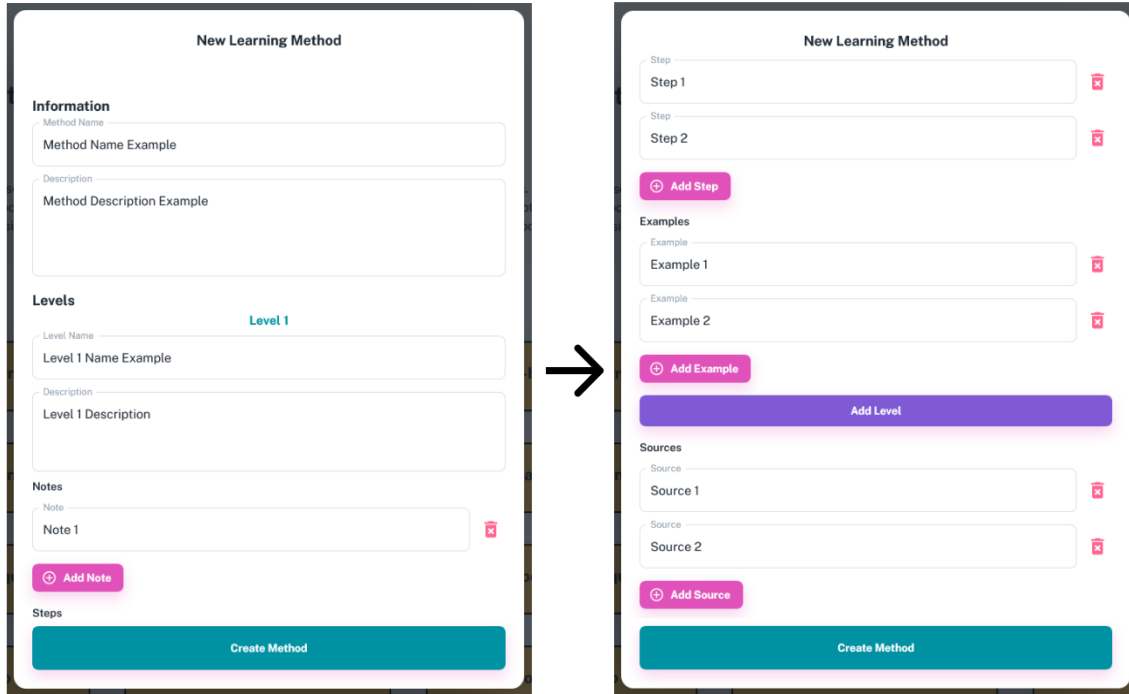

Figure 39. The Learning Methods modal fields

If the user hasn't filled all the fields and tries to submit the new method, the system will alert the user that a field has not been filled and it is required to do so (Figure 40). After the user has filled all the missing information, the learning method will be created.

6.2 Creating A New Learning Object

Selecting an object's category, which might be a video, sound, image, text, quiz, or questionnaire object, is the first step in generating a new learning object. (Figure 25). The development of a questionnaire object will be shown for this presentation because it requires a more complicated process than the production of the other object kinds. Upon clicking the questionnaire object card, users are directed to the form builder playground, as shown in Figure 27. In the event of a user performing a drag-and-drop operation from the panel to the playground field, the form.io modal will be triggered in order to customize the component the user drags (figure 40).
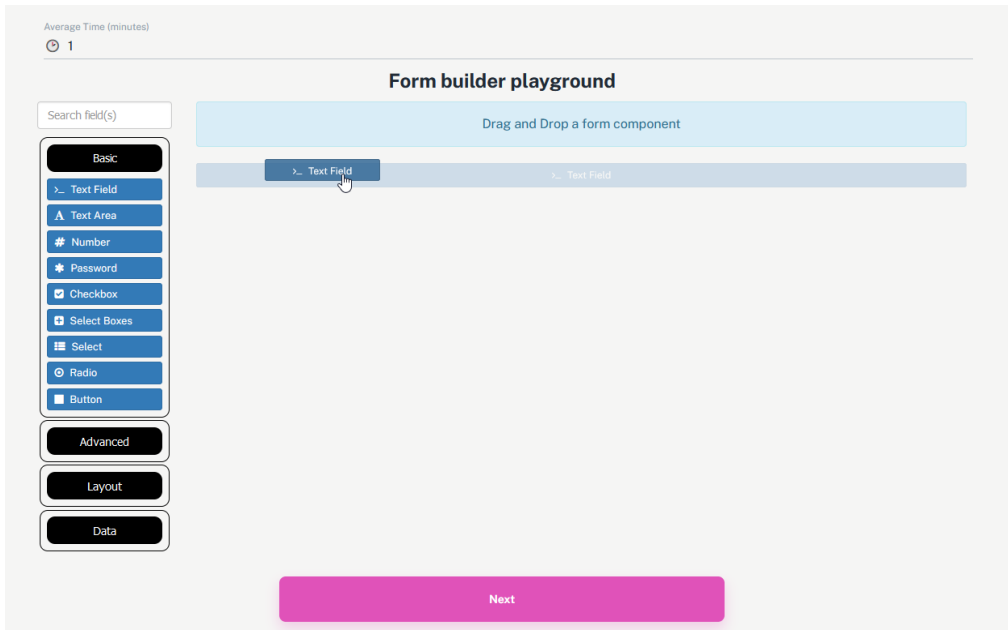
Figure 40. The drag and drop of a text field in the form playground

In the example of the text field, a modal will be opened showing all the options the user can change about that text field component, like the label, placeholder and position and in the top right corner, a preview of the component can be seen (Figure 41).
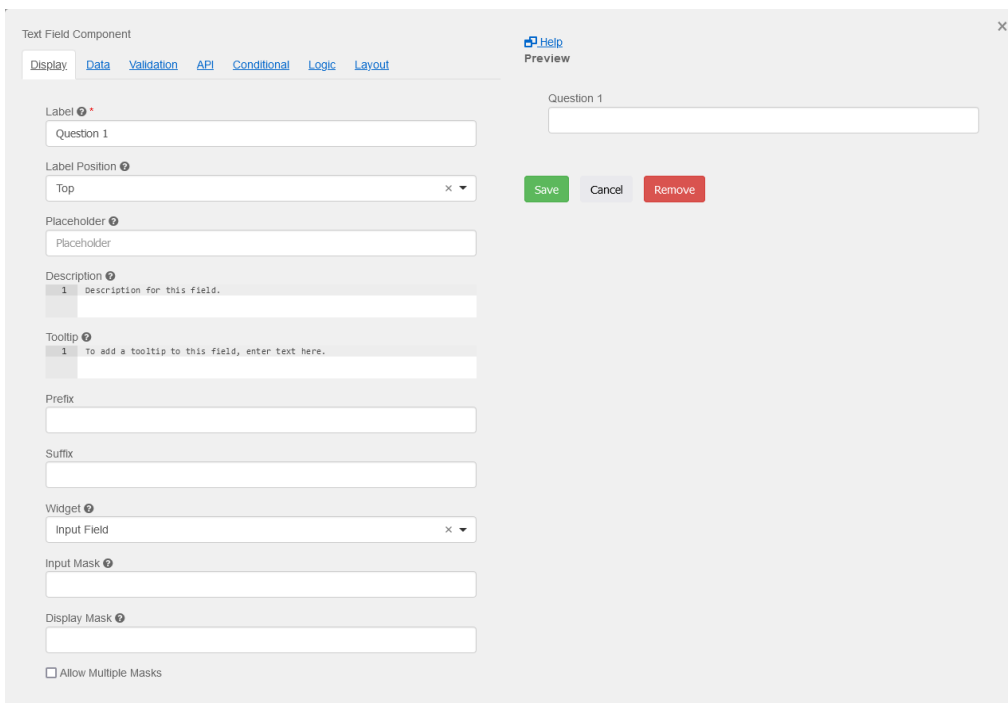

Figure 41. The text field customization modal

49

In the case of a select option component, users have the ability to input answer choices directly into the data tab's designated box. In order for the system to assess whether a student's response is correct or incorrect, they can also assign values to each response. Additionally, the form allows for multiple-answer questions, giving evaluations more flexibility to be tailored to different learning objectives (figure 42).

Depending on whether a field has been filled out or not, the form also allows you to choose whether to show or hide a certain field. With the use of this feature, a questionnaire may be made more dynamic and interactive by changing how the content is presented in response to student feedback. The form builder gives educators and content producers the ability to create engaging and focused evaluations that meet a range of learning requirements and preferences by offering various customizable choices.
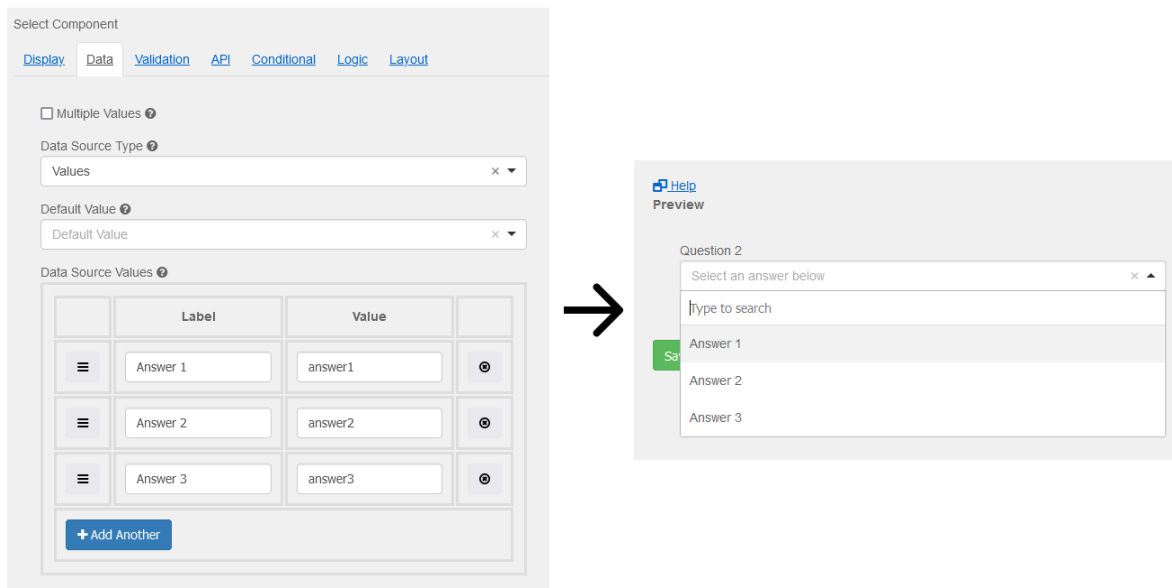

Figure 42. Placing data sources in the select field component

After that the user can proceed to the object tagging step. The object tagging as shown in figure 43, is formed by single and multiple selects components. All the components are required to be selected and the multiple select components are shown with the chip component to indicate the choices the user made. After all the fields are filled, the learning object can be saved.

50

Figure 43. The object tagging with multiple elements

6.3 Creating A Worksheet Skeleton

Creating a worksheet skeleton involves selecting numerous text fields and select components. Users must fill out the worksheet skeleton's basic information, including its name, description, and learning objective, in the first phase. The description field can hold as much data as the user thinks is required, enabling a thorough overview of the worksheet's content and function.

The user's selections in earlier fields are used to dynamically filter each field. When a learning subject is selected, for instance, the courses, educational levels, and educational grades will automatically be filtered to show possibilities that are pertinent. The user experience is improved by this interactive feature's streamlined selection process and assurance that all options are appropriate given the context. An error will be generated for that particular field if a user tries to move on to the next stage with an empty field or clicks on a field and leaves it blank, as seen in Figure 44. By ensuring that all necessary data is provided before continuing, this validation process encourages the development of thorough and well-organized worksheet skeletons.

Figure 44. The Worksheet Skeleton error example

During the evaluation objects selection phase, users are required to choose six evaluation objects for each dimension of Bloom's taxonomy. The "Next" button will be initially disabled upon entering this step and will not become active until the validation method returns true, as shown in Figure 34. The assessment object tables are concealed within accordion components in order to improve user experience and avoid confusion. (Figure 45). Users are not overwhelmed by the whole amount of alternatives thanks to this design decision, which makes it simple for them to access and select the appropriate assessment objects for each dimension. The interface supports a more streamlined and user-friendly choosing process by organizing and structuring the information.

Figure 45. The Evaluation Objects tables

In the learning objects step selection, the user has to select at least one object for each goal he chose from the previous step (Figure 35). The loading bar proceeds as the user click the "Next Goal" button. The next button will enable once the users has filled the validation (Figure 46).
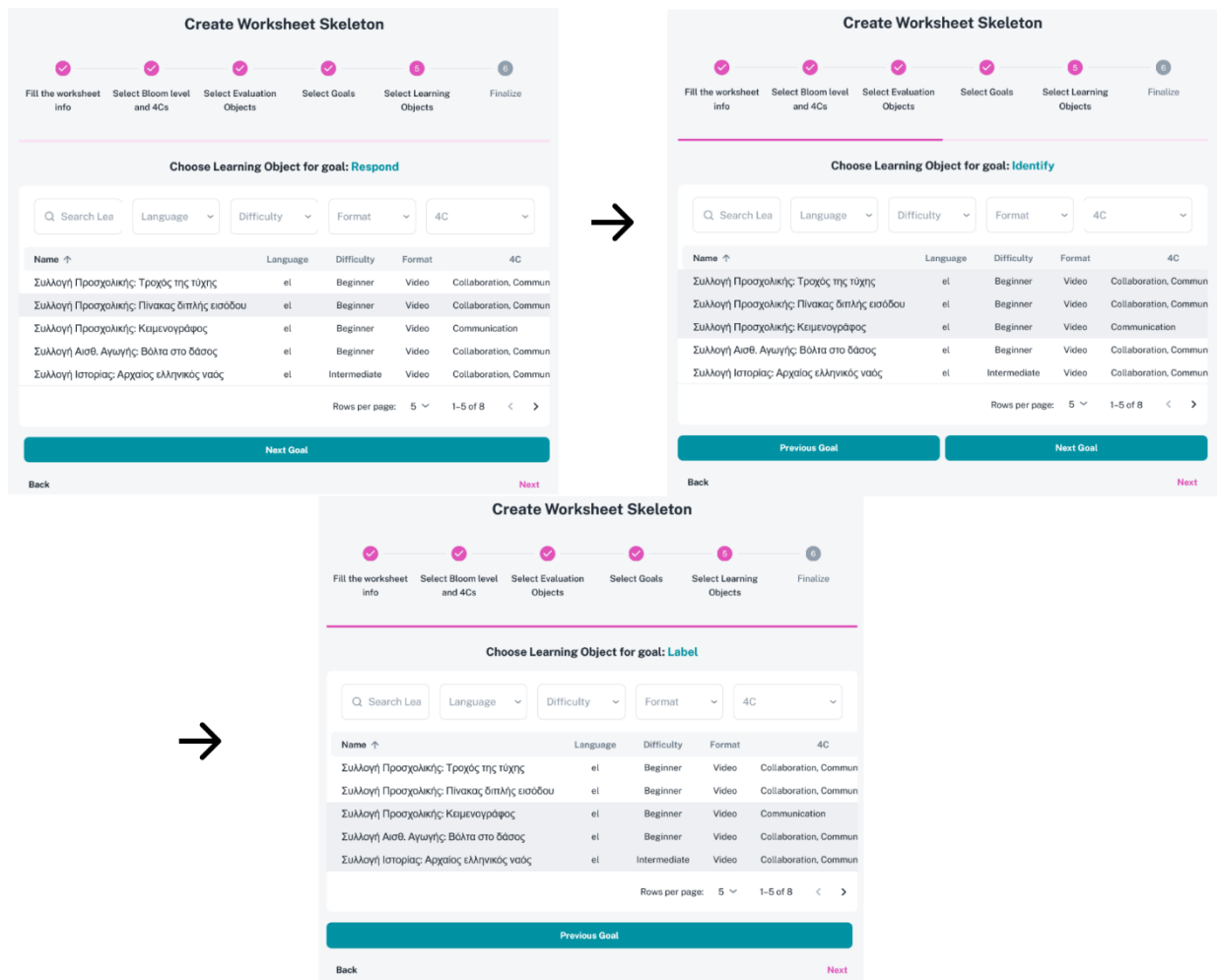
Figure 46. The validation for Learning Objects

The final step in the process presents a summary of all the selected fields for the worksheet skeleton. Users can switch between accordions to view the data presented at each stage, as shown in Figure 47. Below these accordions, the aims, evaluation criteria, and learning objectives are listed.

The objectives are arranged according to their corresponding Bloom's taxonomy aspects. Users can switch between learning and assessment objects as necessary thanks to the listing of the associated evaluation objects directly beneath each dimension. Before completing the worksheet skeleton, users may simply examine and confirm their options thanks to the presentation's organization. Upon clicking the "Finish" button, a backdrop will appear, signaling that the data is being sent to the server and processed. This visual cue provides users with real-time feedback on the status of their submission, ensuring a smooth and transparent experience.
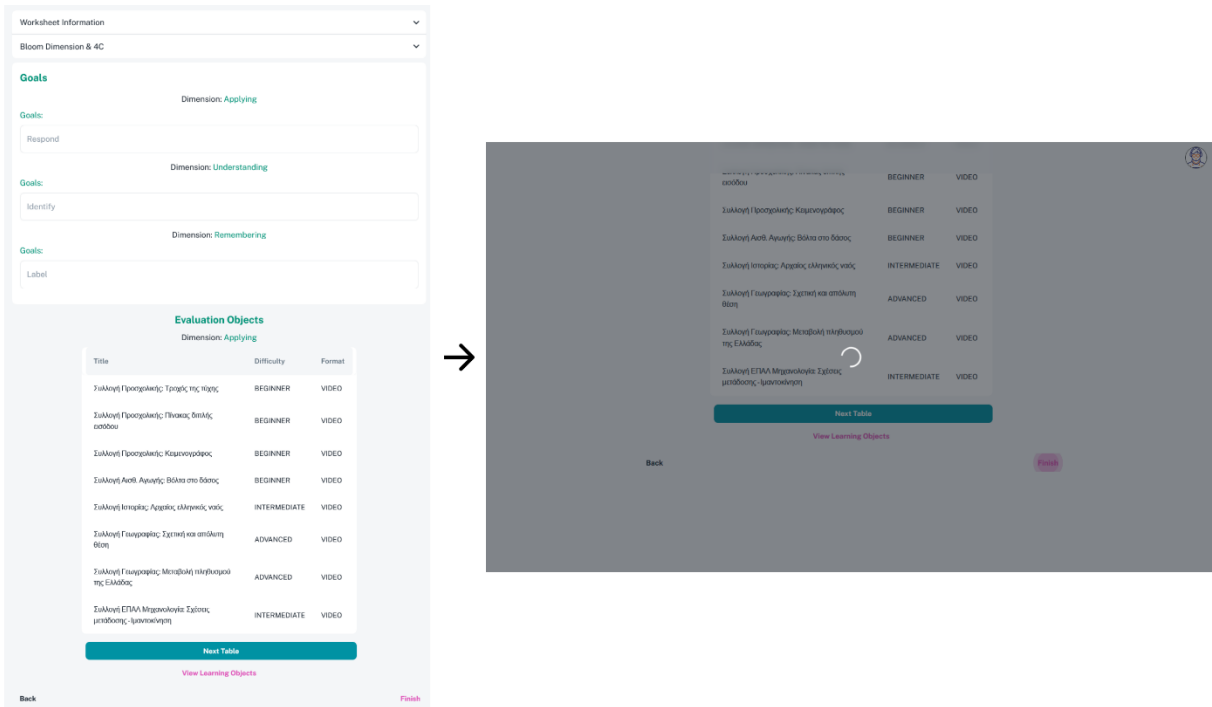
Figure 47. The final step with the backdrop component

6.4 Settings Panel

The settings panel consists of the dark mode option, together with the localization options. The sun and moon icons represent the state of the theme mode. Pressing the moon icon will turn on the dark mode for the whole page. The language options are shown in a select box (Figure 48).
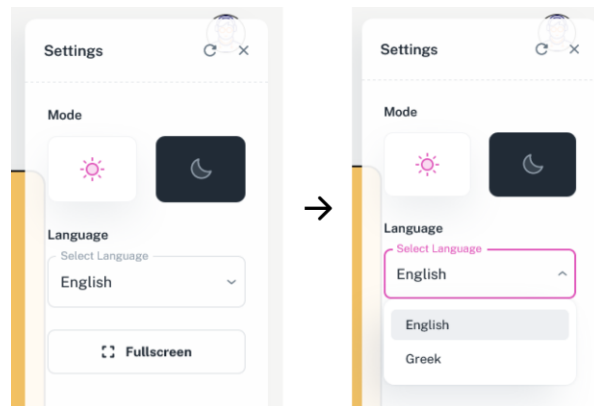
Figure 48. The Settings panel.

Activating the dark mode will results in completely makeover of the GUI using darker colors on the same color palette. The color of all the menu cards is swapped to purple with deep

black font colors (Figure 49). This will satisfy the user and reduce the eye strain in low-light conditions. Additionally, it can present a completely new user interface if the user will be disinterested of the light mode. Last but not least, the localization switch can be seen, where all the English fonts are replaced with Greek.
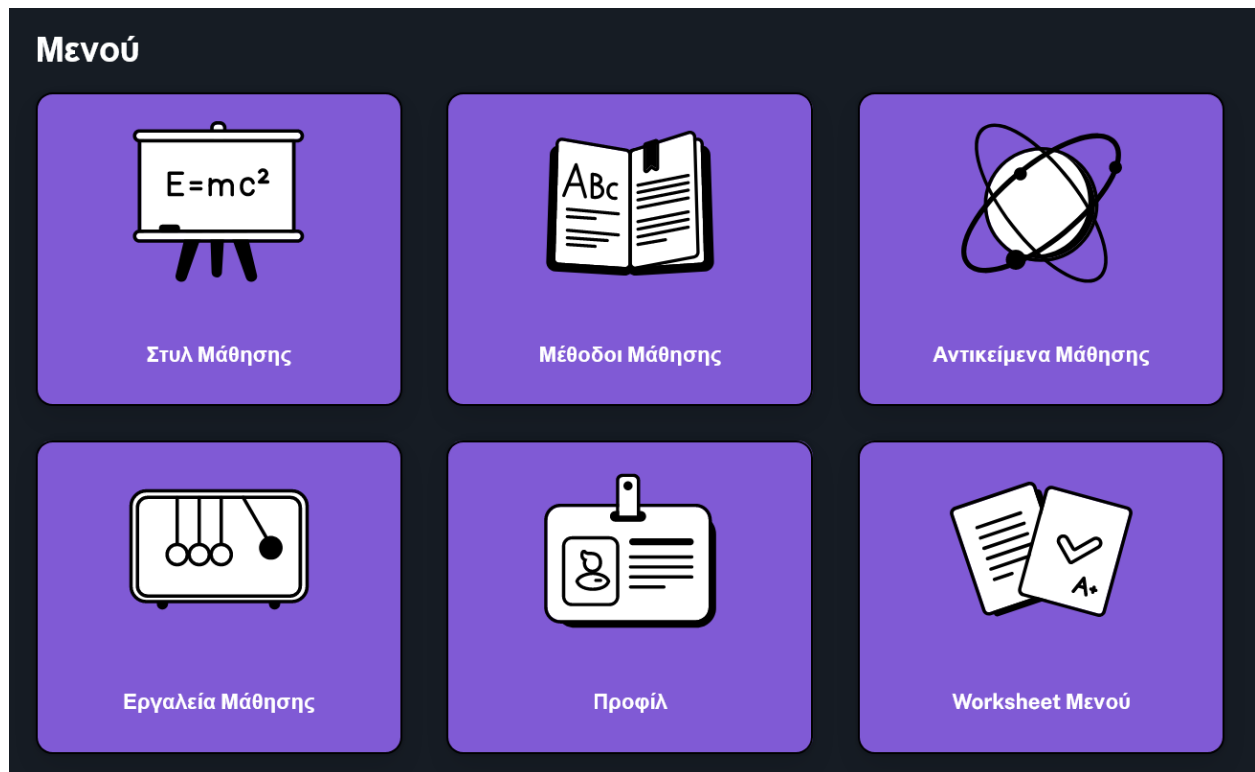
Figure 49. The activation of the dark mode and localization

Chapter 7 - Conclusions and future work

In conclusion, this thesis effectively developed and demonstrated a user-friendly interface that provides a favorable user experience for learners. The proposed interface has shown its potential to engage learners and support their educational journey by carefully analyzing the principles of good UI design, adding kid-friendly colors and design elements, and emphasizing simplicity, clarity, and ease of use.

By emphasizing the crucial importance of UI design in developing interesting and useful learning environments, this research adds to the general knowledge of e-learning and its position in the future of education. It is crucial to address UI design issues and create cutting-edge solutions that address the various needs of students all over the world as e-learning continues to develop and assume a larger role in contemporary education

Future study could look into other ways to improve the user experience and adapt to the always changing educational landscape by building on the successful design and development of a user-friendly interface for e-learning systems. Customizing user interfaces to meet each person's particular learning style would result in a more individualized and successful educational experience. This is one possible avenue for future research.

Future efforts should concentrate on strengthening inclusivity and accessibility in the UI design to make sure that e-learning platforms can accommodate users with a range of requirements and abilities. This can entail creating functions like scalable text, image alternative text, and screen reader compatibility. E-learning systems can offer a more inclusive learning environment for all users by solving these accessibility issues.

Another potential area of future research is the integration of third-party Learning Management Systems (LMS) into e-learning platforms. Users can gain from a cohesive learning environment that unifies course materials, assessments, and progress tracking across many platforms by establishing compatibility between existing LMS solutions and e-learning platforms. While ensuring that e-learning platforms remain flexible and compatible with a wide range of LMS systems, this integration would make it possible to manage learning experiences for both learners and educators in a more efficient manner. This partnership may result in better user experiences, better learning results, and a rise in the use of e-learning platforms in a variety of educational contexts.

References Or Bibliography

[1] A. M. Maatuk, E. K. Elberkawi, S. Aljawarneh, H. Rashaideh, and H. Alharbi, "The COVID-19 pandemic and E-learning: challenges and opportunities from the perspective of students and instructors," *J Comput High Educ*, vol. 34, no. 1, pp. 21–38, 2022.

[2] E. H.-K. Wu, C.-H. Lin, Y.-Y. Ou, C.-Z. Liu, W.-K. Wang, and C.-Y. Chao, "Advantages and constraints of a hybrid model K-12 E-Learning assistant chatbot," *Ieee Access*, vol. 8, pp. 77788–77801, 2020.

[3] S. Kumar Basak, M. Wotto, and P. Belanger, "E-learning, M-learning and D-learning: Conceptual definition and comparative analysis," *E-learning and Digital Media*, vol. 15, no. 4, pp. 191–216, 2018.

[4] E. G. Dada, A. H. Alkali, and D. O. Oyewola, "An investigation into the effectiveness of asynchronous and synchronous e-learning mode on students' academic performance in National Open University (NOUN), Maiduguri Centre," *International Journal of Modern Education and Computer Science*, vol. 11, no. 5, pp. 54–64, 2019.

[5] K. Y. Chau, K. M. Y. Law, and Y. M. Tang, "Impact of self-directed learning and educational technology readiness on synchronous E-learning," *Journal of Organizational and End User Computing (JOEUC)*, vol. 33, no. 6, pp. 1–20, 2021.

[6] C. G. Ogbonna, N. E. Ibezim, and C. A. Obi, "Synchronous versus asynchronous e-learning in teaching word processing: An experimental approach," *S Afr J Educ*, vol. 39, no. 2, pp. 1–15, 2019.

[7] V. Arkorful, N. Abaidoo, and others, "The role of e-learning, advantages and disadvantages of its adoption in higher education," *International journal of instructional technology and distance learning*, vol. 12, no. 1, pp. 29–42, 2015.

[8] I. Yuhanna, A. Alexander, and A. Kachik, "Advantages and disadvantages of Online Learning," *Journal Educational Verkenning*, vol. 1, no. 2, pp. 13–19, 2020.

[9] I.-C. Tudorache, A. M. Iordache, M.-T. Iordache, and others, "The advantages and the disadvantages of e-learning in European union," *J Inform Syst Manage*, vol. 6, no. 2, pp. 1–7, 2012.

[10] V. Handayani, F. L. Budiono, D. Rosyada, R. N. S. Amriza, S. U. Masruroh, and others, "Gamified learning platform analysis for designing a gamification-based ui/ux of e-learning applications: A systematic literature review," in *2020 8th International Conference on Cyber and IT Service Management (CITSM)*, 2020, pp. 1–5.

[11] H. Joo, "A study on understanding of UI and UX, and understanding of design according to user interface change," *International Journal of Applied Engineering Research*, vol. 12, no. 20, pp. 9931–9935, 2017.

[12] H. S. JOO, "A Study on UI/UX and Understanding of Computer Major Students," *International journal of advanced smart convergence*, vol. 6, no. 4, pp. 26–32, 2017.

[13] M. Indriana and M. L. Adzani, "UI/UX analysis & design for mobile e-commerce application prototype on Gramedia. com," in *2017 4th International Conference on New Media Studies (CONMEDIA)*, 2017, pp. 170–173.

[14] K. V Vlasenko *et al.*, "UI/UX design of educational on-line courses," in *CTE Workshop Proceedings*, 2022, pp. 184–199.

[15] E. Krisnanik and T. Rahayu, "UI/UX integrated holistic monitoring of PAUD using the TCSD method," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2273–2284, 2021.

[16] H. Ji, Y. Yun, S. Lee, K. Kim, and H. Lim, "An adaptable UI/UX considering user's cognitive and behavior information in distributed environment," *Cluster Comput*, vol. 21, pp. 1045–1058, 2018.

[17] L. I. Bilousova, L. E. Gryzun, and N. V Zhytienova, "Fundamentals of UI/UX design as a component of the pre-service specialist's curriculum," 2021.

[18] H. S. JOO, "A Study on UI/UX and Understanding of Computer Major Students," *International journal of advanced smart convergence*, vol. 6, no. 4, pp. 26–32, 2017.

[19] D. P. Kristiadi *et al.*, "The effect of UI, UX and GX on video games," in *2017 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, 2017, pp. 158–163.

[20] H. Joo, "A study on understanding of UI and UX, and understanding of design according to user interface change," *International Journal of Applied Engineering Research*, vol. 12, no. 20, pp. 9931–9935, 2017.

[21]   S. V Kolekar, R. M. Pai, and M. P. MM, "Adaptive user interface for moodle based E-learning system using learning styles," *Procedia Comput Sci*, vol. 135, pp. 606–615, 2018.

[22]   A. Hidayat and V. G. Utomo, "An Architecture of adaptive online module system based on Felder-Silverman learning style model," in *International Conference on Online and Blended Learning 2019 (ICOBL 2019)*, 2020, pp. 70–73.

[23]   D. Zlatković, N. Denić, M. Ilić, and A. Zakić, "Providing dynamic adaptivity in Moodle LMS according to Felder-Silverman model of learning styles," 2022.

[24]   T. Y. Wang and C. H. Wang, "E-learning platform of STEAM aesthetic course materials based on user experience," in *2018 1st International Cognitive Cities Conference (IC3)*, 2018, pp. 123–128.

[25]   L. M. Hasani, D. I. Sensuse, R. R. Suryono, and others, "User-centered design of e-learning user interfaces: A survey of the practices," in *2020 3rd International Conference on Computer and Informatics Engineering (IC2IE)*, 2020, pp. 1–7.

[26]   I. Katsaris, I. Logothetis, K. Katsios, and N. Vidakis, "Adaptive Blended Learning Platform based on the 4Cs Architecture.," in *CSEDU (2)*, 2022, pp. 251–259.

[27]   React official website, "React" https://react.dev/learn/thinking-in-react (accessed April. 2, 2023).

[28]   Ibadehin Mojeed "What is the virtual DOM in React?", Aug 16, 2022, https://blog.logrocket.com/virtual-dom-react (accessed April 2, 2023)

[29]   Material UI official website, "MUI" https://mui.com (accessed April. 2, 2023).

[30]   React Hook Form official website, "React Hook Form" https://mui.com (accessed April. 2, 2023).

[31]   Joseph Mawa, "React Hook Form vs Formik - Comparing the most popular React form libraries", Jan 30, 2023 (accessed April 2, 2023)

[32]   Vite official website, "Vite" https://vitejs.dev/ (accessed April. 2, 2023)

[33]   Vite official documentation, "Why Vite" https://v3.vitejs.dev/guide/why.html (accessed April 2, 2023)

[34]   Typescript official website, "What is TypeScript?" https://www.typescriptlang.org/ (accessed April 2, 2023)

[35] Typescript: Documentation, "TypeScript for JavaScript Programmers?" https://www.typescriptlang.org/docs/handbook/typescript-in-5-minutes.html (accessed April 2, 2023)

[36] React-i18next official documentation, "What is react-i18next?" https://react.i18next.com/#what-is-react-i18next (accessed April 2, 2023)

[37] Form.io Documentation, "Forms and Resources" https://help.form.io/intro/overview#forms-and-resources (accessed April 2, 2023)