Adaptive Learning Web Application to Improve CAD Learning in Engineering

Gabriel Cerna Pedro Victor[1], Antoni Perez-Poch[2], Francisco Alpiste Penalba[3] and Jordi Torner[4]

Universitat Politècnica de Catalunya. EEBE, Av. Eduard Maristany, 16 08019 Barcelona. España,
[1] pedro.victor.gabriel@upc.edu, [2] antoni.perez-poch@upc.edu, [3] francesc.alpiste@upc.edu,
[4] jordi.torner@upc.edu

Corresponding author: Pedro Victor Gabriel, pedro.victor.gabriel@upc.edu

Abstract. Adaptive Learning using AI is crucial for improving students' comprehension of computer-aided design (CAD) subjects in engineering and adapts to adaptive learning styles by providing personalized content in a dynamic learning environment using advanced algorithms and data analytics. This system improves students' performance and satisfaction in learning CAD systems. Its components include hierarchically organized content, author dashboards, progress tracking, and gamified exercises that adapt to users in real-time. It employs visual and text representations to express content to help learners create three-dimensional models effectively. The system addresses educational challenges by providing content that is individualized to each student's abilities. The article suggests that the AI-based web review system with a gamified web environment helps students master the skills needed to understand and use CAD systems, resulting in higher grades on graphical expression and improved student satisfaction measured using an SEQ survey.

Keywords: Adaptive learning, Computer-aided design, Engineering education, Web-based applications.
DOI: https://doi.org/10.14733/cadaps.2025.498-509

1 INTRODUCTION

Adaptive learning based on a computer learning environment is a methodology that allows the personalization of the path through learning content, adapting it based on student interaction and their prior knowledge [1].

The web review system is Artificial Intelligence - based (AI-based) [2] with a gamified web environment [3, 4], which enhances learners' engagement, while activities appropriate to the student's level of knowledge have been proposed [5–8].

Regarding adaptive learning, there is a growing use of data collection and analysis technologies, such as educational data mining, learning analytics, machine learning, and artificial intelligence, implementing algorithms to identify student behavior. The objective is to personalize and adapt teaching and learning processes [9, 10].

An adaptive learning system can be supported by different artificial intelligence tools, such as decision trees, Fuzzy Logic (FL), Neural Networks, Bayesian networks, genetic algorithms, and Hidden Markov Models [11–13], which will allow for the diagnosis of the level of knowledge and learning style to adjust the content to the needs of each student. AI has the potential to simulate the decision-making processes of students [14–16].

One of the main challenges faced by students who use Computer-Aided Design (CAD) systems is acquiring and reinforcing specialized skills and industrial standards that enable them to create three-dimensional parts more effectively [17]. Given that each student develops these skills in a highly individualized manner, they require tools that support them in this process and are tailored to their needs.

To address this issue, this article proposes the use of a web-based review system that adapts to the needs of students using Artificial Intelligence (AI) engines, as described by several authors [2, 11, 15]. This tool has been tested and it has been proven that its use improves the grades of students studying Graphic Expression in Higher Education.

The system utilizes EventSource technologies [18] in combination with heuristic systems to produce a predictive algorithm that can be adapted in a personalized manner to the student presenting the content is adjusted to their cognitive needs. Consequently, the proposed system can provide a personalized agenda for each student based on the concepts required for the use of CAD systems.

The adaptive learning web application designed to enhance the learning of Computer-Aided Design (CAD) subjects in Engineering is a valuable resource for students and educators. This tool aims to improve the overall learning experience by adapting individual learning styles, offering personalized feedback, and providing a dynamic and engaging learning environment. By utilizing advanced algorithms and data analytics, the application can identify areas where students may need additional support and tailor the learning experience accordingly.

In conclusion, the system offers a tailored learning experience for students using CAD systems, which can help them acquire and reinforce necessary skills and industrial standards more effectively. The tool is designed to cater to the individual needs of each student, providing a personalized agenda optimized for their cognitive abilities. The use of adaptive learning web applications has been shown to improve student outcomes, increase student engagement, and enhance the overall effectiveness of CAD education.

2 PROPOSED METHOD

The objective of the system is to provide students with the necessary knowledge and skills to utilize Computer-Aided Design (CAD) systems, encompassing both theoretical concepts and specialized aspects of CAD systems.

An experimental approach is implemented to evaluate the effectiveness of the proposed method. The data collected during the academic year 2021-2022 at the Diagonal-Besòs Campus of the Universitat Politècnica de Catalunya – BarcelonaTech (UPC) was used for this purpose.

The adaptive learning system comprises various components, as illustrated in Figure 1, including hierarchically organized content in categories and subcategories, content creation panels, student progress tracking panels, and gamified systems that present exercises and adapt them in real-time to the user. As indicated by the authors cited in the aforementioned article [17], there is a growing need for educational systems to adapt to the rapidly changing technological landscape.

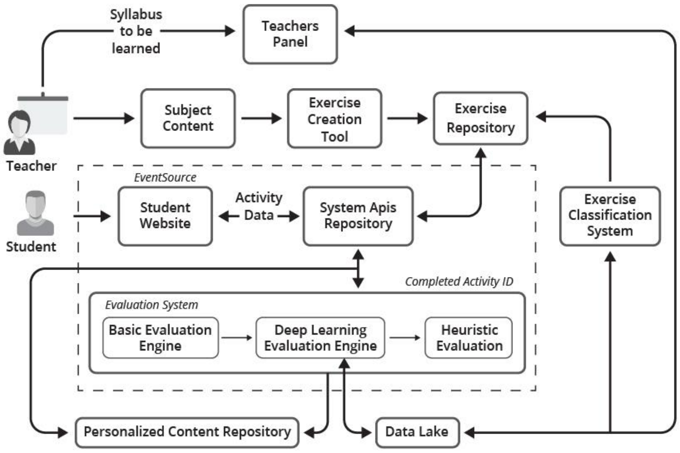

Figure 1: Structure of internal components of the adaptive learning web system.

Before a student uses the system, it is essential to classify and present the content to be learned in visual exercises that offer multiple representations, such as Button Panel, Button Panel with Audio, Button Panel with Time, Button Panel with Text, Memory, Relate, Drag and Drop Text, Drag and Drop Order, Reading, What's Missing, Word Search, Grid, Puzzle, Word Count, Reading and Questions, Reading Speed, Drag Drop Sentences, Drag Drop Groups, Speaking and Cloze Text. These various forms of representation are referred to as engines.

The engines have been grouped to focus on fundamental aspects of understanding Computer-Aided Design (CAD) systems. The Visual Pattern Recognition Engines, for instance, aid in the visual comprehension of drawings and 3D parts, as depicted in Figure 2 and the engines of attention and understanding of the regulations that involve the creation of 3D prototypes such as ISO and UNE standards see Figure 3.

The goal is to cover all the tasks associated with the direct manipulation of 3D object creation software, such as SolidWorks and Rhinoceros.

The delegation of expressing the content correctly falls under the purview of the teachers, who are responsible for imparting instruction in the following sub-subjects: Regulations, Geometry of space, surfaces, and visualization of sketches, with a particular focus on the use of software such as SolidWorks.

Figure 2: Drag and drop for work visual container.

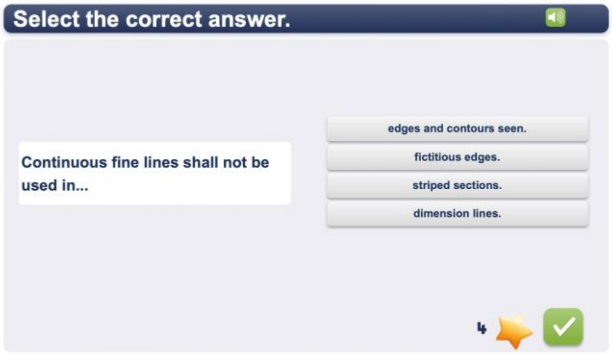

Figure 3: Example of exercise for work normative topics.

As a result of the creation of content, we were able to generate 1001 exercises distributed as follows:

Number of Exercises	Type of Representation
430	Button panel
80	Button panel with text
84	Matching
86	Drag and Drop
82	Alphabet Soup
84	Puzzle
76	Word Count
78	Drag & Drop Groups
1	Close Text

Table 1: Distribution of exercises according to presentation type.

The syllabus contents were divided by sub-subjects. Then, the content that could be learned independently from other parts of the sub-subjects was identified. This hierarchical categorization allowed the creation of levels of exercises linked to each other that work on specific topics, such as the competence with which the student reviews normalization issues, the identification of dihedral representations, and the correct way of constructing objects CAD.

The CAD content used in this system is the one used in the subject of Graphic Expression, which teaches the use of CAD programs such as SolidWorks,

In addition to the categorization of the curriculum content as explained above, for us, it was important to establish three dimensions in which the student would benefit: topic improvements in the use of the CAD program, interpretation of parts to be developed in CAD programs, and international regulations related to the mass production of objects generated in CAD programs.

For the first point, the engines that were used the most were puzzles, drag and drop, and matching because they allow visual improvement in the association of the use of CAD programs.

For the second point exercises, puzzles, drag and drop, matching, and button panels were used because they allow the development of sequences in which the student must develop the pieces or aspects that must be taken care of in their 3D modeling. Finally, in the third case, the most used engines were button panels, Drag and Drop, Matching, and Alphabet Soup to reinforce the theory as well as the aspects of the use of the technical drawing components used in the development of 3D drawings.

The method used to improve the understanding of CAD systems and the knowledge associated with them is that the student performs five activities per day for no more than 17 minutes.

Each time the student completes each activity, the student's activity is corrected in the evaluation system, and the next content to be presented in the next session is determined based on the effectiveness obtained. If the effectiveness is less than 60%, the student is asked to repeat the level with exercises that are marked as simple. If this behavior is repeated, the AI system will recommend that the student download another topic that it infers is more appropriate. If the student's performance is very low, in addition to the recommendations of the AI evaluator, a heuristic engine is run that looks at the student's overall performance in the other sub-subjects and decides whether the student should be dropped in more than one sub-subject, if it is observed that there are problems in more than two other sub-subjects.

The duration of 17 min is the result of an experiment described in [17], in which the students performed the daily activities corresponding to each of the above-mentioned sub-subjects. The average time spent by the students per day was 17 minutes, and the time per activity was 4 minutes, except in activity one, which is a mental warm-up and whose exercises are not directly related to the subject to be studied, only aims to introduce the student to the various types of engines that will be found in each question with an average duration of 50 seconds.

The 17 minutes of daily activities applied to students are within the time frame in which users can have active attention, as explained by Bradbury in his article [19].

The complexity of the content is adjusted automatically, depending on the performance of the previous activity; if this activity has a low or very low performance, it will be repeated for the student, but with a lower level of complexity, to ensure that the student understands the concept of learning or review. On the other hand, the web system adapts the complexity in real time to the exercises of the activity that the student is performing, which aims to increase or decrease the complexity, as shown in Figure 4.

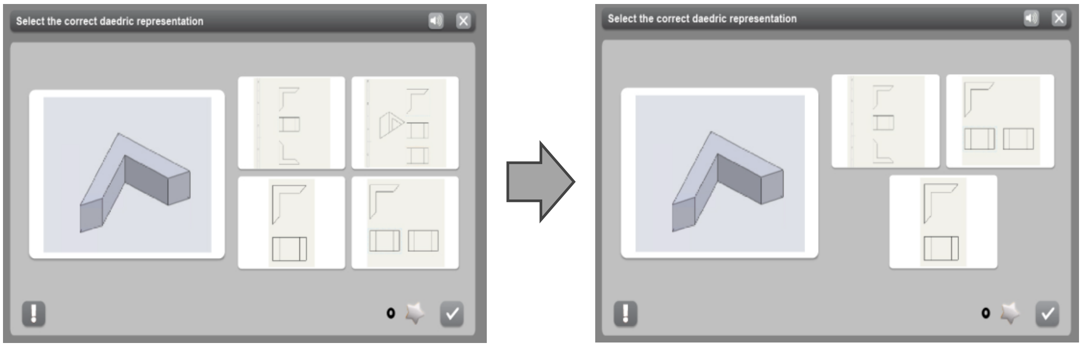

Figure 4: Example of how an exercise can be made easier in real-time**.**

Once each activity has been performed, the results are sent to a task manager to be sent to the evaluation process, which allows the system not to collapse owing to the overuse of users. The task queue allows the system to evaluate the activities independently of the rest of the components of the adaptive web system, and this allows the service to be always active regardless of the workload assigned to it and reduces the loss of data produced by any communication interruption between the servers and the web system, Figure 5 shows a diagram of how this microservice was implemented.

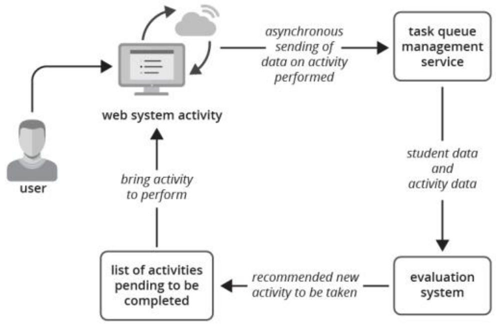

Figure 5: Flow describing how the system processes the results of the activity performed by the user.

The evaluation system operates in three phases: the first one is a general evaluation, which evaluates what the student has done with respect to what the average of students who have done the activity, then it proceeds through a process of deep learning that works with a neighborhood analysis algorithm (KNN), as described in [20–23]. Upon completion, it looks to see if the student has problems in other areas of the syllabus or other sub-subjects and recommends a general level change if that is the case, Figure 6.

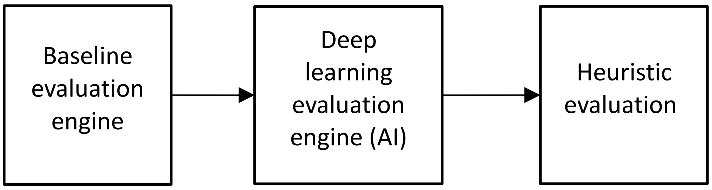

Figure 6: Steps of the evaluation system.

2.1. Baseline Evaluation Engine

The purpose of this phase is to determine if the student has successfully completed an activity, taking into account the number of successes and errors made in the time of the activity, using the average of the exercises performed by the students throughout the last year, in this case the school year 21-22.

The evaluation equation (2.1) of this phase uses the MId parameter, defined as the average number of exercises performed at that level by students in the previous year.

$$eff = \frac{\sum Suc - \sum Fail}{MId} \tag{2.1}$$

Where eff is the score of the level performed, Suc is the correct answer, and Fail is the incorrect answer.

However, during the experiment conducted in the EEBE, we found users who performed more than once the total number of exercises in each activity or in some cases, the activity required more time in its realization and did not perform all the exercises of the activity, this type of behavior was detected in the activities that required visually identify dihedral representations of 3D objects to be performed in CAD systems

To solve this, we created a component called cycles, described by equation (2.2). This allowed us to adjust the calculation of the initial grade for each activity.

$$cycles = ceil\left(\frac{CE}{MId}\right) \tag{2.2}$$

The final evaluation equation for eff if cycles is more than 1, is as follows the equation (2.3):

$$eff = \frac{\begin{gathered}\left(\sum_{i=0}^{ceil(MId)} Suc_i - \left(Kr * \sum_{i=0}^{ceil(MId)} Fail_i\right)\right) + \\ \left(\sum_{j=ceil(MId)+1}^{2*ceil(MId)} 0.9 * Suc_j - \left((Kr - 0.2) * \sum_{j=ceil(MId)+1}^{2*ceil(MId)} Fail_j\right)\right) \\ + \left(\sum_{k=2*ceil(MId)+1}^{CE} 0.8 * Suc_k - (Kr - 0.4) * \sum_{k=2*ceil(MId)+1}^{CE} Fail_k\right)\end{gathered}}{CE} \tag{2.3}$$

Where Kr is a number that allows adjusting the penalty for each error, to simulate what is implemented in the online exams of the subject and that after the tests performed, the most optimal value was 1. The values 0.2, and 0.4 that multiply Kr, mimic the current evaluation of the exams, in which each time an exercise is performed repeatedly, each failure has greater relevance, and instead for each success made more than once has a lower weight and CE is the number of exercises performed by the student.

Translated with DeepL.com (free version)If the evaluation is less than 6, the system assumes that there is some kind of problem with the student, which can range from a simple oversight to the student not having the basics to take that content in all cases the system allows the user to perform the same level twice but with the difference that each time the exercises that were categorized as less difficult are presented, this categorization is done once a month, using the statistical data of use of each of them.

2.2. Deep Learning Evaluation Engine

The purpose of this evaluation phase is to determine whether the students who have similarities to the current student have the same behavior, as proposed by phase one, otherwise learn and recommend the next activity to learn.

This is done by employing an algorithm for neighborhood analysis. Before the implementation of the KNN-based evaluation engine, the following models were tested: Support Vector Machine (SVM), Random Forests (RF), Bayes Naive Classifier (BNC), and K-Nearest Neighbors (KNN). Although SVM, RF, and BNC perfectly understand the relationships between the input variables and the leap to recommend, the problem of adjusting the recommendation based on the user's acquired knowledge at that time was one in which the KNN model proved to be more useful.

The system was developed using the K-Nearest Neighbors (KNN) algorithm, as described by Zhang in detail in [20]. This algorithm was chosen because it requires minimal data to begin learning and identify similarities between users. To recommend the next level for a learner (Ui), the model must consider the behavior of other users who have shown similar patterns in the last four levels, where the number of levels to be considered in the vectors is k.

During the analysis and training of the model, k was tested from one to five consecutive levels, with three and four being the models that had the best accuracy and k =4, which had an accuracy closest to 1.

The metrics with which the KNN model has been evaluated are balanced accuracy, the accuracy and the confusion matrix resulting from a summary of the predictions made by the model and organized in a table by categories.

To implement this evaluation engine, vectors of variables were created: time spent performing the level, number of correct exercises, number of incorrect exercises, efficiency achieved in the level, branch-course-level structure, difficulty of the exercises performed in the level, difficulty of the exercises performed in the next level by other users with similar patterns, and GPS coordinates where the activity is performed.

2.3. Heuristic Evaluation

The function of this phase is to reposition a student with problems in several learning paths of different subsubjects to a lower level.

A condition to consider is that when creating the agenda that the user must learn or review, time points are marked within each sub-subject and assigned to the paths, which can be, for example, the change in content type, change in complexity of the content, or a change of course. These points are used by the system as a reference to move a student's path, as described below.

The system uses as a base the current level that is being evaluated; if it has an effectiveness of less than 60% and the student has repeated it consecutively more than twice with problems, then all the learning paths of the other sub-subjects are analyzed; if in more than three evolutionary paths of the student, an effectiveness of less than 60% is detected and they are from two other different sub-subjects, the system forces a generalized descent of the user to a point previously marked in the definition of the paths, such as a change of competence or at the beginning of a previous course if the system contains more than one concatenated course, such as graphic expression 1 and graphic expression 2.

3 RESULTS

The experiment conducted aligns with research by Gabriel Cerna as outlined in [17]. This study involved two separate groups: one group utilized an adaptive web system and the other did not.

The current experiment differs from the previous experiment in several ways. First, the maximum time allowed for each session was increased to 17 min, and it consisted of five activities. Second, a distinction was made between visual engines and text-based exercises that focus on improving theoretical comprehension. Finally, a more precise categorization of the difficulty level of the exercises was introduced, based on the usage patterns of the users in the course of 2021-2022.

The implementation of the method described in Figure 1, which can be accessed at the URL https://egedao.tech, has been completed. The tests were conducted over the course of academic years 2021-2023, with 708 enrolled students divided into 24 class groups. Each group had a maximum of 32 students, and was scheduled either in the morning or afternoon.

Four of the morning groups were selected to participate in the web tool trial, referred to as the "experimental groups." These groups were chosen because they had previously participated in the trial audits. However, the assignment of the groups to teachers was not controlled by the researchers; thus, the selection process was subject to a certain degree of randomness.

It is widely recognized that students are allocated to lecture halls based on their preferred order and university entry exam results, making it difficult to compare groups. To address this issue, four additional control groups were selected, whose class schedules matched those of the experimental groups in both day and hour, ensuring that their entry exam results were also similar. It is important to note that none of the teachers in the experimental group taught in the control group.

The six assessments used to evaluate students included the following:

- Practical examination of CAD proficiency and regulatory understanding. This involved the design of components, assemblies, and drawings, accounting for 25% of the total score.
- A practical test on CAD abilities and hands-on experience with Geometry and Surfaces contributed to 15% of the overall score.

- An assessment of freehand drawing skills, accounting for 10% of the score.
- A theoretical examination of technical drawings, comprising 15% of the total score.
- A theoretical test of geometry accounted for 10% of the total score.
- A group project, which accounted for the remaining 15% of the score. The project was make by a team of three students.

The participants in the experimental groups were given the opportunity to take the course using a web-based tool, similar to the other groups taking the course (internal control test). The well-known Pygmalion effect, which is the result of an individual's expectations affecting their performance, was considered in this study. To prevent this, the students in the experimental groups had the option of selecting whether they wanted to complete the course using the web tool or drop out at any time. Furthermore, they were informed that they would receive the higher of the two grades obtained (web tool or internal control test) and that this grade would also apply to the theoretical exams.

Taking the course using the web tool required a higher level of dedication from the experimental groups, as they were limited in the time they could devote daily and weekly to the course. To maintain their commitment and avoid falling behind, it was essential for these students to complete all exercises.

In contrast, the control groups only had the days of the exam to prepare and could study little by little or only on the days immediately preceding the exam. Additionally, neither the students, teachers, nor any other subjects in the control group were aware that the test was being administered.

4 CONCLUSIONS

This research article details the development of an adaptive learning web application designed to enhance computer-aided design (CAD) education in engineering and an assessment of the results of its application in Engineering Education in a Higher Education institution. This system is unique in that it can adjust to individual learning styles and provide personalized feedback to the students. The system utilizes advanced algorithms and data analytics to identify areas where students may require additional support and customize the learning experience accordingly. The effectiveness of the system has been demonstrated through testing, which showed that it improves student outcomes, increases student engagement, and enhances the overall effectiveness of CAD education. The system comprises various components, including hierarchically organized content, content creation panels, student progress tracking panels, and gamified systems that present exercises and adapt in real-time to the user.

In addition to the above-mentioned points, to better understand the results and to be able to reach the conclusion of the experiment, it is necessary to remember that one of the problems that we face is that several teachers of the subject of Graphic Expression participated in the test, this fact makes it necessary to homogenize in some way the grades and ways of grading, to achieve this we used a method proposed by Alpiste et al.[24]

Similarly, a Student Evaluation of Educational Quality Questionnaire (SEEQ) survey was administered to students as described by Grammatikopoulos [25].

The survey assessed nine dimensions related to teaching quality, and our results showed that students had a positive view of the use of methodological-didactic resources and the evaluation of exams and grades. These findings suggest that the students were satisfied with the teaching methods and how they were evaluated in our study.

The use of the SEEQ survey allowed us to collect valuable opinions on the quality of teaching in our study, providing information on the effectiveness of our teaching methods and student satisfaction. The dimensions that carried the most weight during this study were learning, enthusiasm, and evaluation. The results indicated that the use of the adaptive review web tool not only reduced stress related to high theoretical content but also helped students obtain better grades, as it allowed them to approach theory exams in the subject of graphic expression more effectively.

In terms of evaluation results, we observed an 8% improvement in the scores obtained in the nominative theory test and a 6% improvement in the geometric theory test compared to the control groups see figure 7.

To validate the results, a bilateral Pearson correlation was performed using the IBM SPSS Statistics 28.0.1.0 software package on the scores of the experimental group in the Normative and Geometric tests, on those obtained using the web tool, and on the operational data of the web tool.

We limited the data to those showing statistically significant linear associations when the bilateral significance level $p < 0.01$ is met (i.e., the null hypothesis was rejected, and therefore, the correlation was reliable). All these correlations are positive, and the Pearson correlation coefficient values are within the range of "considerable" ($0.5 < rP < 0.75$) or "strong" ($0.75 < rP < 0.9$).

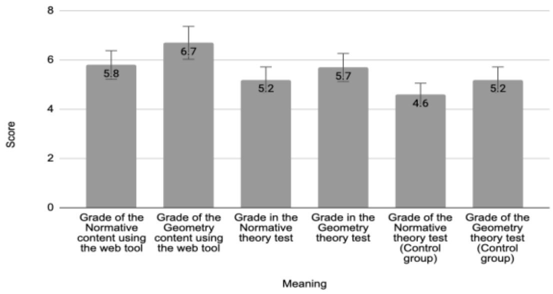

Figure 7: Comparative scores between the groups that used the adaptive system versus the control groups during the years 2021 to 2023.

We can conclude that, although the experiment was limited to a single common core engineering subject, the theoretical exams and, consequently, the final grades of the students who participated in the experiment improved compared to those who did not.

The results were sufficiently satisfactory and positive for the students, and it can be proposed that the number of participants in the following experiments be increased using this method.

This system can be implemented in a range of educational institutions across diverse fields of study. This can be beneficial in other areas of education, where state-level standards govern the material to be learned.

5 FUTURE WORK

This system demonstrates the flexibility to be effectively deployed in a variety of educational institutions spanning diverse academic disciplines. Furthermore, its applicability extends beyond specific fields of study to encompass other realms of education where learning materials adhere to state-level standards. This can be beneficial in other areas of education, where state-level standards govern the material to be learned. For instance, this system can prove highly advantageous in university degrees, such as those in science or humanities, where the curriculum content is similarly structured and regulated.

REFERENCES

[1] van Riesen, S.A.N.; Gijlers, H.; Anjewierden, A.A.; de Jong, T.: The influence of prior knowledge on the effectiveness of guided experiment design. Interactive Learning Environments. 30, 2022. https://doi.org/10.1080/10494820.2019.1631193.

[2] Kahraman, H.T.; Sagiroglu, S.; Colak, I.: Development of adaptive and intelligent web-based educational systems. In: 4th International Conference on Application of Information and

Communication Technologies, AICT2010, 2010. https://doi.org/10.1109/ICAICT.2010.5612054.
[3] Lavoué, É.; Monterrat, B.; Desmarais, M.; George, S.: Adaptive Gamification for Learning Environments. IEEE Transactions on Learning Technologies. 12, 2019. https://doi.org/10.1109/TLT.2018.2823710.
[4] Tenório, K.; Dermeval, D.; Monteiro, M.; Peixoto, A.; Silva, A.P. da: Exploring Design Concepts to Enable Teachers to Monitor and Adapt Gamification in Adaptive Learning Systems: A Qualitative Research Approach. Int J Artif Intell Educ. 32, 2022. https://doi.org/10.1007/s40593-021-00274-y.
[5] Kossack, F.; Bender, B.: individualization in engineering design education: implementation of an adaptive e-learning environment (ade-le). In: Proceedings of the Design Society, 2023. https://doi.org/10.1017/pds.2023.230.
[6] Liu, M.; McKelroy, E.; Corliss, S.B.; Carrigan, J.: Investigating the effect of an adaptive learning intervention on students' learning. Educational Technology Research and Development. 65, 2017. https://doi.org/10.1007/s11423-017-9542-1.
[7] Al-Othman, M.A.; Cole, J.H.; Zoltowski, C.B.; Peroulis, D.: An Adaptive Educational Web Application for Engineering Students. IEEE Access. 5, 2017. https://doi.org/10.1109/ACCESS.2016.2643164.
[8] Martín, E.; Carro, R.M.: Supporting the Development of Mobile Adaptive Learning Environments: A Case Study. IEEE Transactions on Learning Technologies. 2, 2009. https://doi.org/10.1109/TLT.2008.24.
[9] Dogan, M.E.; Goru Dogan, T.; Bozkurt, A.: The Use of Artificial Intelligence (AI) in Online Learning and Distance Education Processes: A Systematic Review of Empirical Studies. Applied Sciences (Switzerland). 13, 2023. https://doi.org/10.3390/app13053056.
[10] Capuano, N.; Caballé, S.: Adaptive learning technologies, 2020. https://doi.org/10.1609/aimag.v41i2.5317.
[11] Almohammadi, K.; Hagras, H.; Alghazzawi, D.; Aldabbagh, G.: A survey of artificial intelligence techniques employed for adaptive educational systems within e-learning platforms. Journal of Artificial Intelligence and Soft Computing Research. 7, 2017. https://doi.org/10.1515/jaiscr-2017-0004.
[12] How, M.L.; Hung, W.L.D.: Educational Stakeholders' independent evaluation of an artificial intelligence-enabled adaptive learning system using bayesian network predictive simulations. Educ Sci (Basel). 9, 2019. https://doi.org/10.3390/educsci9020110.
[13] Pérez-Sánchez, B.; Fontenla-Romero, O.; Guijarro-Berdiñas, B.: A review of adaptive online learning for artificial neural networks. Artif Intell Rev. 49, 2018. https://doi.org/10.1007/s10462-016-9526-2.
[14] Demartini, C.G.; Sciascia, L.; Bosso, A.; Manuri, F.: Artificial Intelligence Bringing Improvements to Adaptive Learning in Education: A Case Study. Sustainability. 16, 2024, 1347. https://doi.org/10.3390/su16031347.
[15] Gligorea, I.; Cioca, M.; Oancea, R.; Gorski, A.-T.; Gorski, H.; Tudorache, P.: Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature Review. Educ Sci (Basel). 13, 2023, 1216. https://doi.org/10.3390/educsci13121216.
[16] Akavova, A.; Temirkhanova, Z.; Lorsanova, Z.: Adaptive learning and artificial intelligence in the educational space. E3S Web of Conferences. 451, 2023, 06011. https://doi.org/10.1051/e3sconf/202345106011.
[17] Gabriel Cerna Pedro Victor, Antoni Perez-Poch, Francisco Alpiste Penalba, Jordi Torner Ribe: An Adaptive Methodology for the Improvement of Knowledge Acquisition by a Multimedia Web Tool. International Journal of Engineering Education. 39, 2023, 643–652.
[18] HTML Living Standard — Last Updated 15 March 2024.
[19] Bradbury, N.A.: Attention span during lectures: 8 seconds, 10 minutes, or more? Adv Physiol Educ. 40, 2016. https://doi.org/10.1152/advan.00109.2016.
[20] Zhang, S.: Challenges in KNN Classification. IEEE Trans Knowl Data Eng., 2021. https://doi.org/10.1109/TKDE.2021.3049250.

[21] Lopez-Bernal, D.; Balderas, D.; Ponce, P.; Molina, A.: Education 4.0: Teaching the basics of knn, lda and simple perceptron algorithms for binary classification problems. Future Internet. 13, 2021. https://doi.org/10.3390/fi13080193.

[22] Maheswari, K.; Priya, A.; Balamurugan, A.; Ramkumar, S.: WITHDRAWN: Analyzing student performance factors using KNN algorithm. Mater Today Proc., 2021. https://doi.org/10.1016/j.matpr.2020.12.1024.

[23] Nugroho, A.; Riady, O.R.; Calvin, A.; Suhartono, D.: Identification of Student Academic Performance using the KNN Algorithm. Engineering, MAthematics and Computer Science (EMACS) Journal. 2, 2020. https://doi.org/10.21512/emacsjournal.v2i3.6537.

[24] Torner Ribé, J.; Brigos Hermida, M.Á.; Alpiste Penalba, F.: informació: Professor Index Requirement (PIR): a standardized method to obtain teacher comparative qualifications based on student scores achievements. In: International Conference on Education and New Learning Technologies. "EDULEARN18, 1–9, Palma de Mallorca, Spain, 2018.

[25] Grammatikopoulos, V.; Linardakis, M.; Gregoriadis, A.; Oikonomidis, V.: Assessing the Students' Evaluations of Educational Quality (SEEQ) questionnaire in Greek higher education. High Educ (Dordr). 70, 2015, 395–408. https://doi.org/10.1007/s10734-014-9837-7.