A Novel E-Learning Platform for Building and Publishing Student-Driven Personalized Lessons

Ayman Hajja
Department of Computer Science
College of Charleston
Charleston, SC, USA
hajjaa@cofc.edu

Austin J. Hunt
Department of Computer Science
College of Charleston
Charleston, SC, USA
huntaj@g.cofc.edu

***Abstract*—In this Innovative Practice Full paper, we will be presenting an instruction-based web-platform that allows instructors to create programmable text- and media-based interactive and personalized mini-lessons.**

The contribution presented in this paper is twofold. First, we provide a flexible and freely-accessible system that allows instructors to effectively create mini-lessons by defining a set of lesson elements, along with the order in which they should appear to students. The second primary contribution is the introduction of a novel personalization component that integrates with our system; this component allows instructors to create lessons that exhibit a "conditional flow", meaning that the order in which lesson elements are presented to students will be guided by some data-driven conditions evaluated using answers provided by students.

Each mini-lesson is comprised of an ordered set of lesson elements. There are four types of lesson elements: the first type is referred to as "information", which can be a paragraph of text, an image, or an audio (or video) segment. The second type of element is a "multiple choice question", which can be either created on-the-fly along with the lesson creation, or randomly selected from a pool of pre-defined questions. The third and fourth types are "label" elements, which serve as location markers in the lesson to which we could branch to, and "jump" components, which can be used to initiate these branches. The flow of these elements are generally sequential; however, the student answers to questions can also be used to affect the flow of these elements, resulting in personalized, student-specific learning paths. At every "question" element, instructors have the ability to define a different branch (or jump) for every possible answer; for example, if one of the answers to a particular "question" element implies a misunderstanding of a topic, then the instructor may specify a branch that will alter the flow of elements' execution such that the next displayed element (or set of elements) are meant to reinforce that topic.

We tested our system using several mini-lessons covering various topics, and we collected data from students regarding their learning experience. The data collected was analyzed and the findings presented in this paper show positive results.

***Index Terms*—personalized learning, e-learning, lesson creation, data-driven**

I. Introduction

In this paper we present an interactive online learning platform built with Django, an open source web framework written in Python. This platform offers a novel lesson format to the e-learning environment that delivers personalization and student engagement through a simple system of instructor-designed lesson flows that vary depending on individual student input. Instructors have at their disposal a toolkit of various types of **components**, the core building blocks of any lesson, that they can quickly and easily create and organize into full lessons of their own creative design. This toolkit leaves it up to the imagination of the instructor to determine how many different paths a lesson can take depending on student input, and whether those paths are only slightly or significantly different from each other. Once published, a lesson presents each student with a lively, feedback-dependent, conversation-like stream of information, questions and answers that may become more and more specific to that student with each answer they provide. Our platform collects submitted answers and offers instructors a **reporting** dashboard for analyzing various metrics related to student understanding of lesson material. While still in development, the platform has been deployed in two Spring 2020 sections of an "Introduction to Computer Organization and Assembly Language Programming" course and continues to grow in its capacity for personalized learning.

II. System Overview

In this section, we will start by introducing the building blocks of our platform: the lesson components (Subsection A). Then, we will describe how these building blocks can be linked and arranged to construct lessons with any desired behavior (Subsections B and C). Finally, we will introduce the reporting and analysis functionality available to anyone who creates a lesson using our platform (Subsection D).

A. Lesson Components

Each lesson is comprised of a set of ordered lesson components; these components can be easily added and re-arranged (through a simple drag-and-drop interface) by instructors to match their desired lesson flow and behavior. We created four types of components that instructors can use as building blocks to construct flexible and customizable personalized lessons. Next, we provide a brief explanation of each of these four components.

Information Components. The purpose of these components is to present text- and media-based segments of information to the student. Each component can be displayed

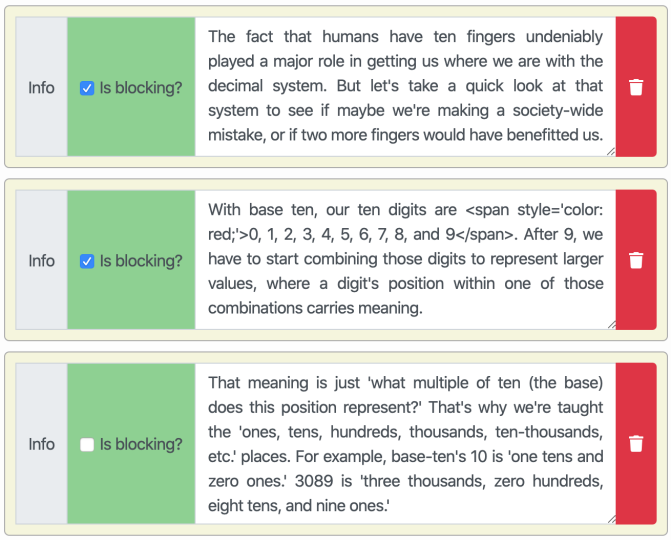


Fig. 1: Example of instructor view of information components

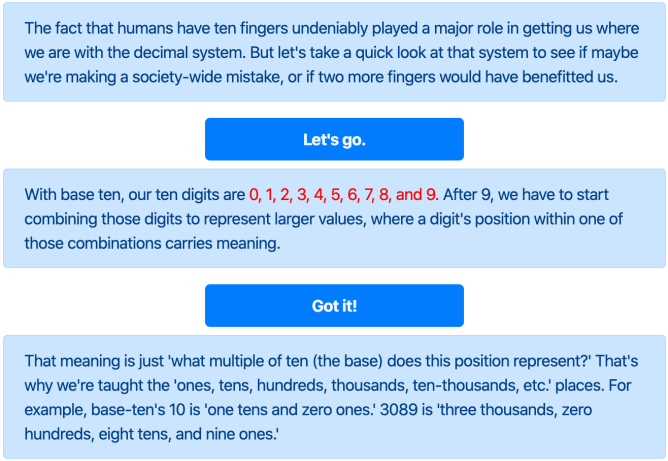
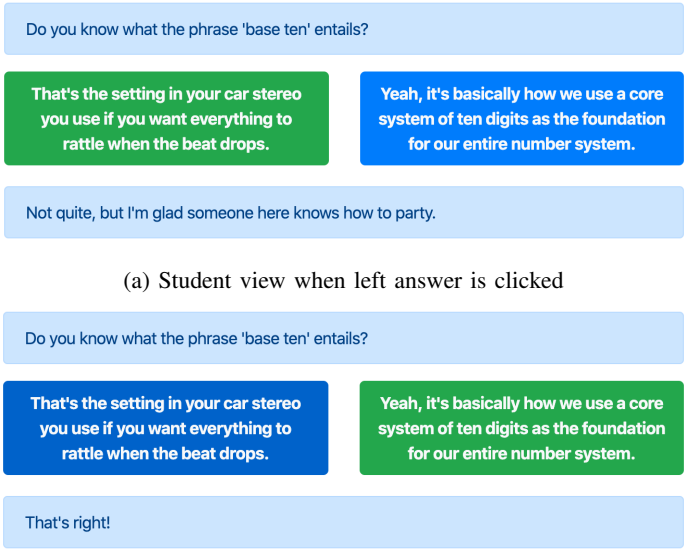


Fig. 2: Example of student view of information components



(a) Student view when left answer is clicked

(b) Student view when right answer is clicked

Fig. 3: Multiple-choice example to increase engagement (student view)

via animation (letter by letter) or all at once. In addition to providing instructors with the ability to customize the general appearance of these components such as font size, font family, and animation speed, we also allow them to decide whether each of these components is a blocking component or not. Similar to a blocking statement in programming, a blocking component will block the execution of the lesson (i.e. halt displaying of components), and wait for the student to click on a "continue" button. Figure 1 shows a screenshot of the instructor view for a section of a lesson that only contains "information components", while Figure 2 shows the student view of that lesson section. The content of these "continue" buttons is pulled from a pre-populated set of labels, which can also be modified by the instructor. The purpose of introducing this blocking feature is two-fold; first, it allows students to move through the lesson at their own pace and without being overwhelmed with information, and secondly, it introduces a way of keeping the student engaged by requesting a confirmation action at short intervals in the form of a click (similar to how streaming platforms ask if you're still there so the content doesn't leave you behind).

Our platform also allows instructors to write HTML code in these information components, which will be parsed by the browser when the lesson is being displayed to the student; as a result, the platform provides flexibility for the type (and visual style) of the content to be displayed. Examples of types and visual styles that have been used in our lessons include subscripts and superscripts, links, colored and underlined portions of text (to emphasize terms), images, and audio files.

Multiple Choice Question Components. As the name implies, these components are used to present a multiple-choice question to the student, followed by a set of answers to choose from. Instructors can provide any number of answers, and our platform will display these answers in a visually appropriate format accordingly. As will be mentioned in the next subsection, instructors can also nest other components inside each answer to provide a personalized experience depending on the students' answers. It is worth mentioning here that although we anticipate most instructors to use "Multiple Choice Question" components for assessment purposes (by providing a legitimate question to test students' understanding), some may use these questions as an opportunity to increase the students' engagement and attention level by incorporating an element of humor; it has been shown in [1] that this can have a positive effect on student learning. Figure 3 shows an example of the latter use.

In a later subsection, we will present the reporting dashboard that our platform provides for instructors, which can

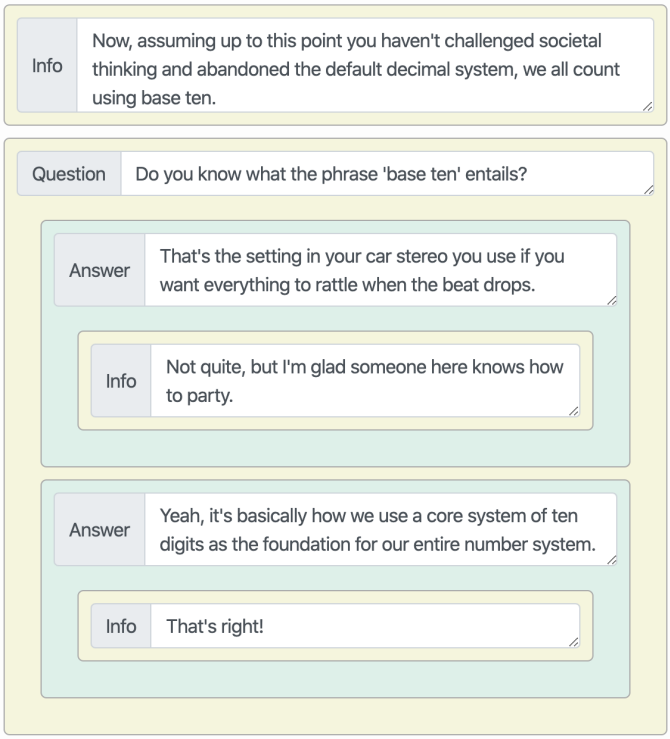


Fig. 4: Multiple-choice example to increase engagement (instructor view)

be used to display summaries of previous student attempts. The answers to these multiple-choice question components are particularly important to the reporting functionality of our platform. Because some of the lesson questions may be used only for engaging the student and not for actual assessment, we allow instructors to easily specify the questions that should be ignored when generating the final report. Figure 4 shows a screenshot of the instructor's view for the multiple-choice question component that was used to generate the student's view in Figure 3. Other options that instructors can specify include whether to shuffle the set of answers or not, and marking the correct answer (which will also be used in generating the lesson report).

In addition to allowing instructors to construct their own multiple-choice questions while creating their lessons, we also allow instructors to define pools of predefined questions that the platform can randomly choose from. This is particularly useful when the desired behavior of the instructor is to loop through a section of the lesson that contains one (or more) of these random multiple-choice question components. We will provide an example in the following subsection that uses this type of question along with the "jump" and "label" components introduced next.

Jump and Label Components. We have decided to combine the "jump" and "label" components together since they function in conjunction with one another. Label components are used to mark a location in the lesson to which we will be

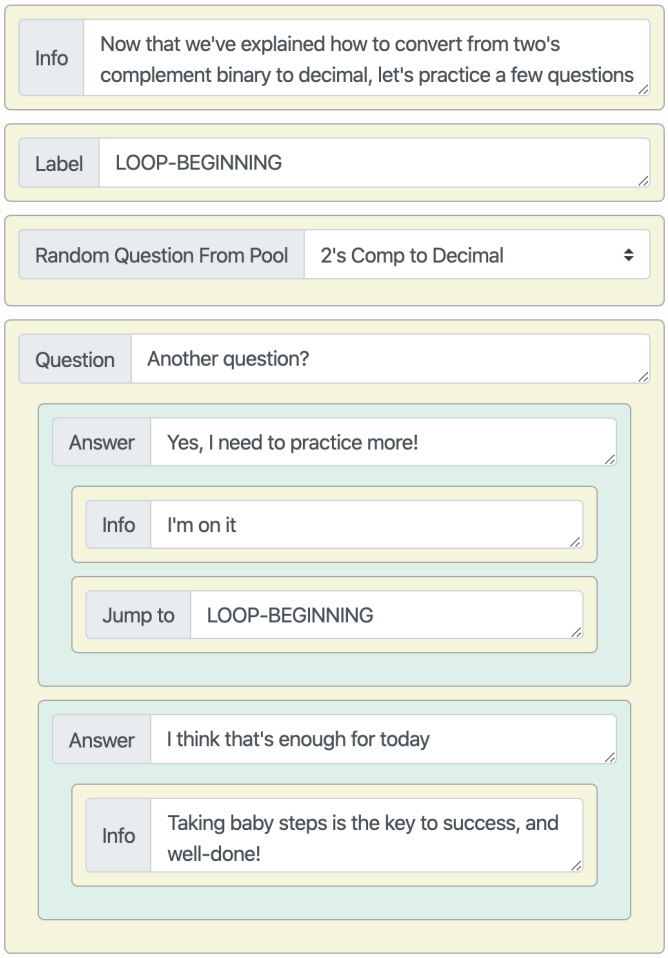


Fig. 5: Lesson section that allows students to keep requesting questions from a pre-generated pool

jumping, and a jump component will simply use a pre-created label to change the flow of the lesson and start the execution from the location of that label.

Figure 5 shows the instructor view for a lesson that will keep prompting students with an additional question for as long as they keep requesting. Figure 6 shows the student view for that lesson for a student that was asked two questions from the same pool upon their request.

B. Lesson Creation

The interface for creating lessons allows instructors to add components, change properties of components, delete components, and through the use of an intuitive drag-and-drop interface, easily change the order of lesson components, and thus the flow of the lesson. In our platform, we adopted an indentation-based component layout that provides clear visual clues as to how these components will be displayed to the student, and in which order.

For example, by glancing at Figure 5, we can quickly notice the following:

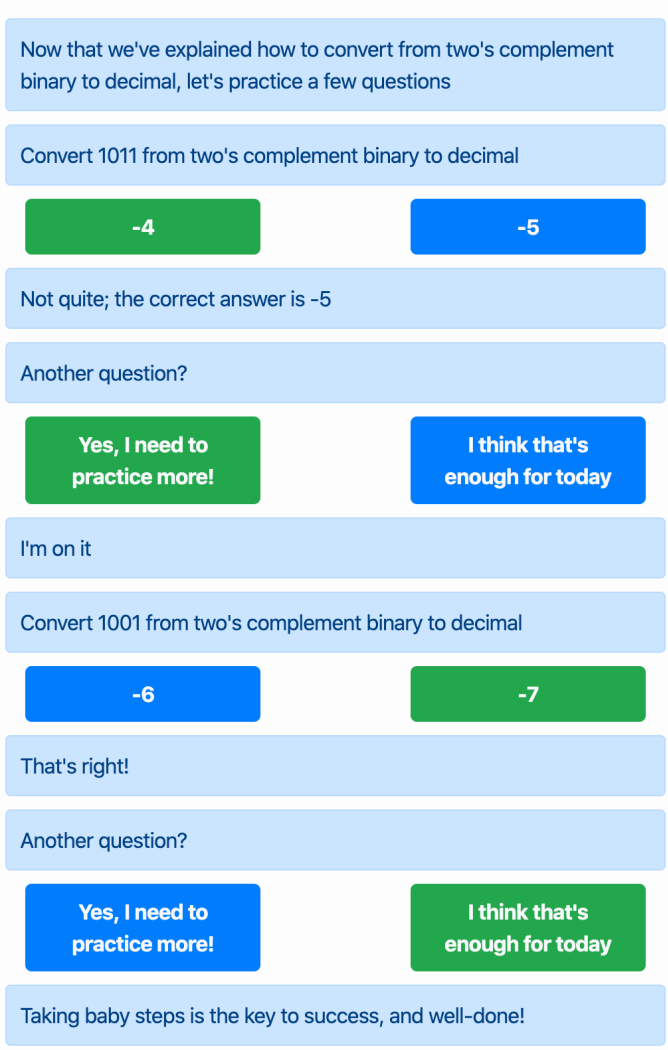


Fig. 6: Student requesting to be asked two questions about the same topic (and from the same question pool)

- The information component ("Now that we've...") will be displayed first, followed by the creation of a label (nothing will be displayed to the student for a label; it's purely a marker for the system), then prompting the student with a question pulled from the pool of questions: "2's Comp to Decimal". Notice here that these three components exhibit the same level of indentation, and therefore they will run sequentially (from top to bottom).
- The "Another question?" question will only be displayed after the student answers the random question from the pool, since all questions are blocking components by default and they will wait for the student to provide an answer.
- Once the student answers the random question from the pool, the "Another question?" question will be asked to determine whether we should prompt the student with another random question or not. Based on the answer to that question, we will either:
  - prompt the student with "I'm on it", then jump back to right before the "Random Question" component to repeat the segment and ask another question, or
  - prompt the student with "Taking baby steps is the key to success, and well-done!", followed by what appears after the "Random Question" component.

Based on the set of components presented in the previous subsection and the example provided above, one should notice that instructors are able to create lessons that exhibit the three types of operation flows necessary to create any desired behavior. These three types of operations are sequential, conditional, and iterative; for example, "information" components that have the same level of indentation will be presented sequentially, "multiple choice questions" introduce a conditional behavior, and the "jump and label" components can be used to introduce an iterative flow. These three types of operations resemble the three necessary flows possible in a programming language (top-to-bottom code execution, if statements, and loops). This deliberate decision was made to provide instructors with enough flexibility to create any desired lesson flow.

C. *Component Nesting and Personalization*

So far, we have presented a few examples that demonstrate how components can be nested within each other, but we have not provided an elaborate discussion on the motivation and significance of nesting components, which we will attempt to do in this subsection.

It should be clear that not all component types can (or should be allowed to) contain other components; for example, and for obvious reasons, an "information" component cannot contain other components. A "question" component on the other hand must contain "answer" sub-components. Note here that we did not previously define a separate "answer" component, since an "answer" is inseparable from a "multiple choice" component; an answer cannot exist without a question, and a question cannot exist without at least a single answer.

The answer sub-component can contain children (i.e. other components defined within it), but it does not have to. What makes this structure even more interesting is the fact that our system allows instructors to place additional questions within an "answer" sub-component. We believe that nesting questions within answers of other questions is particularly significant to both the learning experience and the personalization element of this platform.

There are two primary advantages that instructors can benefit from by utilizing nesting questions:

1) **Assessing the effectiveness of instructions.** Often, a student's answer to a particular question implies a specific misunderstanding (e.g. the correct answer can be obtained by applying two stpng, but the answer provided by the student is the output of only applying the first step). Since we believe that we know what

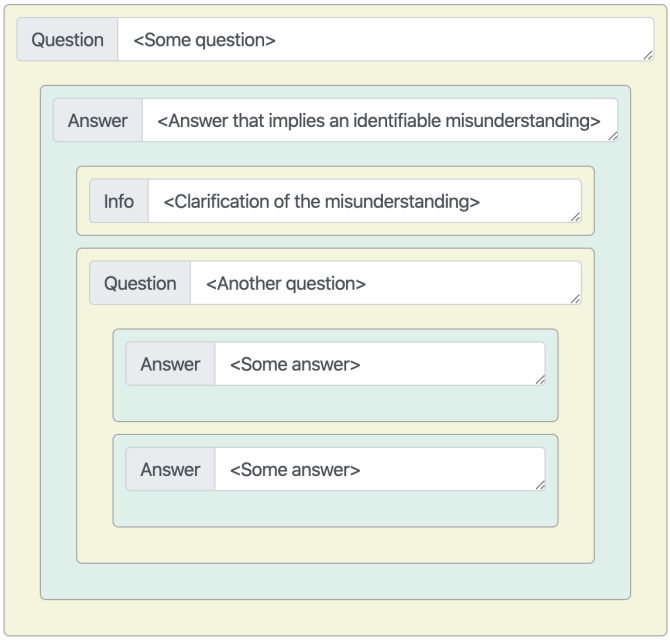


Fig. 7: A use-case of using nested questions for assessing the effectiveness of instructions

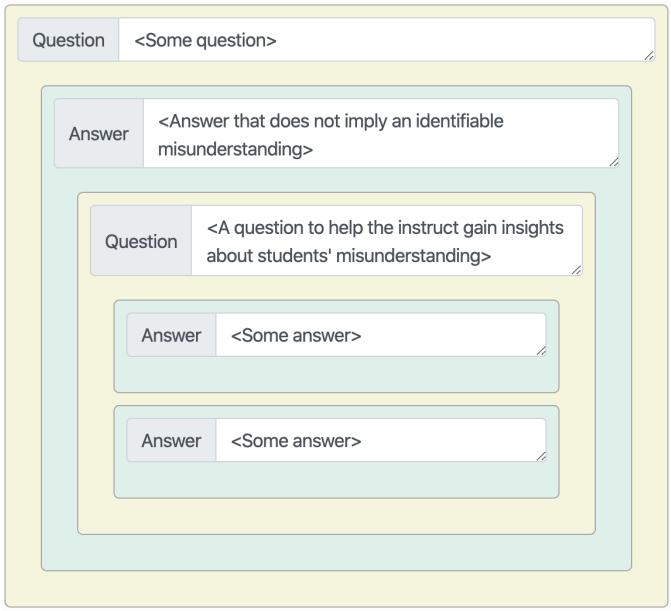


Fig. 8: A use-case of using nested questions for gaining insights about student misunderstanding

the misunderstanding was, instructors can provide a child (information) component for that specific answer to explain what the mistake was. Although this can be done without nesting two questions within each other, the effectiveness of the explanation provided can only be gauged by inspecting the answers provided for a following similar question. Figure 7 shows an example of a lesson (with placeholders) to illustrate this idea. As will be shown in the next subsection, percentages of correct answers for each question can be obtained by instructors; these values can be used to measure the effectiveness of an attempt to clear students' misunderstanding.

2) **Gaining insights about student misunderstanding.** When one is unable to identify a misunderstanding based on a particular answer, it would be invaluable to try to understand how students reached that incorrect answer, since this information can provide insights into students' thought processes. This can be done by also adding a (nested) question component inside one of the answers. Figure 8 shows an example of a lesson (with placeholders) to illustrate this idea.

Because of the importance of creating lessons that exhibit many levels of depth, we built our platform in such a way that allows instructors to build lessons with an unlimited number of nested questions.

D. Reporting

The ability to quickly and easily collect and analyze data is arguably one of the most valuable elements of any e-learning environment. Our platform shines a spotlight on data collection and analysis in the form of a powerful-yet-simple Reporting dashboard that captures insight at both the high and low levels of student learning. The Reporting dashboard essentially offers, for each lesson, a mirrored view of the lesson creation interface, except that the editing tools for each of the lesson's components are replaced by helpful Reporting tooltips indicating statistics about the component. High-level tooltips capture class-[1] and lesson-wide metrics like correct and incorrect ratios, average question response times, lesson view counts, and a summary of the lesson's degree of personalization[2]. Low-level tooltips reveal similar ratio and time metrics on a per-question basis, as well as helpful breakdowns of the individual personalized learning paths. It should be noted here that when calculating the average response times for both per-lesson and per-question analysis, our platform will automatically exclude outlier response times to ensure meaningful data is presented in the Reporting dashboard.

Instructor Summary The highest level of the Reporting dashboard provides instructors with an interface that enables them to monitor how effectively they are teaching. Summary statistics outline the various metrics about the full set of lessons they have created and the class which has viewed them. A given instructor is shown an overall lesson view count accompanied by class-wide averages for metrics like question response time and correct and incorrect answer ratios. Instructors can also choose filters to apply to these

[1]A class in this context is the group of all students that have taken at least one lesson created by a given instructor

[2]The degree of personalization for a lesson is represented by the lesson's depth of nested questions

calculations, such as whether or not to include lessons currently marked as private, or whether to exclude or only include lessons about a certain topic. Suggested topic filters are dynamically provided using automatic keyword parsing against the lessons.

Ratio Analysis. One of the core metrics that drive the Reporting dashboard is the ratio of correct and incorrect answers recorded for a given lesson, or for a given component within a lesson. At the lesson level, our platform allows instructors to sort their lessons by their corresponding ratios of correct and incorrect answers to help them identify lessons containing material that is more commonly misunderstood. A lesson with a close-to-perfect correct ratio is one that likely needs no reinforcement (depending on how large the sample size is, of course), while a lesson with a correct ratio below 70% or 60% is one whose material likely requires additional explanation.

The Reporting dashboard allows instructors to dig deeper by offering, for each lesson, the same basic interface used for lesson creation, only with each component's editing tools replaced by Reporting tooltips containing statistics about the component. As depicted in Figure 9, an instructor can expand and walk through a given lesson to see what the correct and incorrect ratios are for each of the questions that were asked so that they can identify questions or question categories requiring additional instruction.

When it comes to questions that are nested within another question's answers, this platform's front-running idea of learning personalization presents itself heavily in this ratio analysis. As an instructor, viewing these ratios as they become more nested in the personalization tree[3] can provide insight about how effectively things are being explained. If we take the nested question structure in Figure 9, we can see that the first outer question has a low correct ratio, which means most students chose the incorrect answer. The key here is that the information component nested in the wrong answer is acting as the transition to the next question. If that information component contains a sufficient explanation about what was asked, the next question which is similar should have a much higher correct ratio, and indeed, it does. This is how ratio analysis can be used to validate a specific explanation or clarification.

To give an example use case of ratio analysis, take Professor Z, who sees that their lesson on C to MIPS Assembly Translation has a correct ratio of 90% while a lesson on Pointer Arithmetic in C has one of 70%. They also see that both lessons were taken by 20 students. At this point, Professor Z could identify which question or questions in the Pointer Arithmetic lesson caused the low ratio. If those were nested questions, they could analyze the personalized paths to determine whether or not sufficient clarifications were given.

[3]You can think of learning personalization in a tree format; the root is the beginning of the lesson. Every multiple choice question is a node that splits into N branches, where N is the number of choices

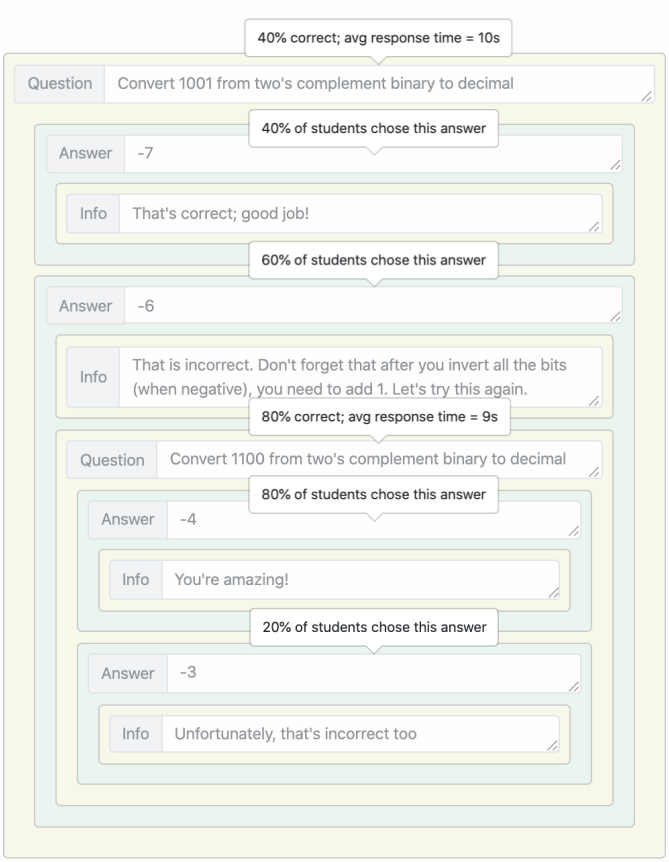


Fig. 9: Using nested question ratio analysis to validate that a second clarification is sufficient after a wrong answer is submitted

If not (i.e. if a clarification was not provided or the second question after a clarification had the same or lower correct ratio), they could create a follow-up lesson expanding on that specific material.

Response Time Analysis. Another big focal point of the work presented here is that it makes it very easy to provide ample opportunities throughout each lesson for students to reflect on material. For every answer submitted by a student in response to a question, our platform saves the amount of time that has passed between the question being asked and the student answering. The lesson-wide average for this metric provides an indicator of how much time, in general, students spend reflecting on that lesson's questions. The Reporting dashboard offers the ability to easily sort lessons by this metric to identify those which may require more attention. If this metric is higher for one lesson than others, that stands as an indicator that students are spending more time reflecting on the questions being asked in that lesson than in others. This could mean that the material is a bit more difficult than that in other lessons, that more instruction needs to be provided, or perhaps that the questions generally just require more thought than those in other lessons.

On a lower level, the Reporting dashboard also offers a view of per-question response time averages, included in the aforementioned Reporting tooltips. This view enables the instructor to identify whether certain questions or question categories need reinforcement with more instruction within the lesson or possibly need to be rephrased. Of course, it is up to the instructor to determine what the response time threshold should be for a given question, as certain questions simply require more reflection than others. As an example, if one question has an average response time of 10 seconds while another or most others are between 3 and 5 seconds, the instructor could gauge the report and determine if the question should be asked differently or perhaps preceded by additional information components.

III. Survey Results

In the Spring semester of 2020, this platform was deployed in two sections of an "Introduction to Computer Organization and Assembly Language Programming" course at the College of Charleston. The students were provided with a set of lessons on the platform and were subsequently surveyed about their overall learning experience during those lessons. The survey prompted the students to indicate a level of agreement for each of a series of statements about the platform ranging from "Strongly Disagree" to "Strongly Agree". The statements included in the survey were designed in such a way that they would produce insight about the platform as it relates to important elements of personalized learning. First, we ask about their perceived level of engagement relative to traditional textbook learning; our platform aims to make the learner feel connected and in charge of their learning through a learner-driven lesson format. Second, we ask about their perceived level of entertainment relative to traditional textbook learning; one of the key elements to this platform is to provide a playful and creative way to teach and to learn. Even the seemingly boring or complex subjects can be better learned if presented in a fun light. Third, we ask them how personalized they believe our system to be - the kicker question, as learning personalization is perhaps the biggest goal behind the platform. Fourth, we ask them how they perceive the role of frequent questions in retaining information; the platform's question-centric lesson format aims to produce a greater level of retention than is found with traditional text-based formats. Lastly, we ask them how they perceived the lessons in the context of reflecting; the proposed lesson format encourages the inclusion of ample opportunities to reflect on lesson material (via the semi-frequent pauses, or "blocking" components discussed previously, or simply Q & A).

The results, as seen in Figure 10, were largely positive. For each of the statements shown, a majority of the 27 total respondents indicated that they either agreed or strongly agreed. Since the statement with the largest ratio of disagree responses references the perception of personalization in the lessons, it is important to note here that the lessons given to the respondents did not utilize the full nesting capability of the system described in Section II-C; that feature was not fully implemented at the time the survey was distributed.

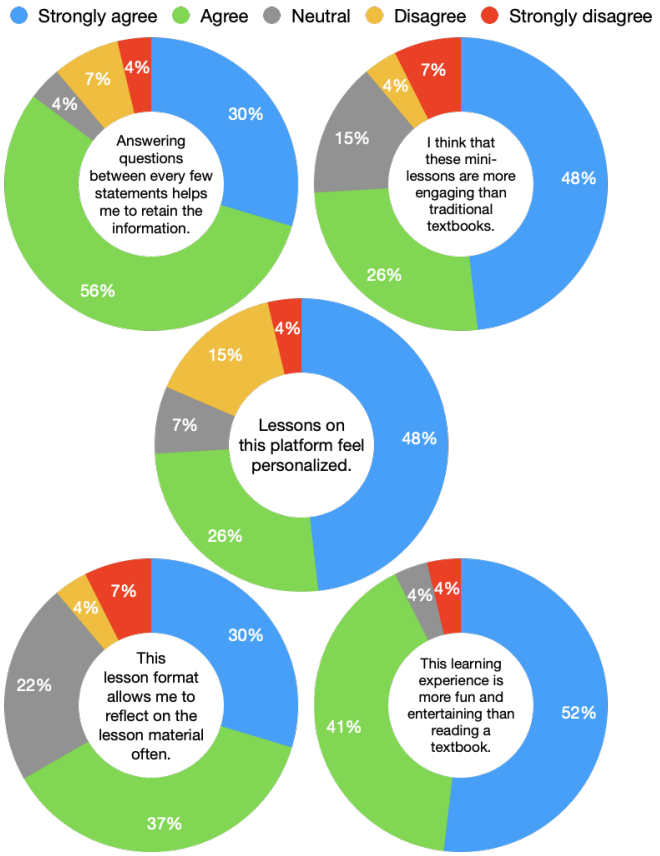


Fig. 10: Results from a student survey about their learning experience

IV. Related Work

Web-based education platforms that utilize gamified lesson formats via "student response systems" similar to our platform have been shown to be effective in improving student learning [2] as well as in achieving student outcomes outlined by ABET[4] criteria for accrediting engineering programs under a continuous improvement model [3].

Also, existing e-learning solutions ThoTh Lab [4] and Pathway [5] integrate elements of personalization (our platform's primary target) for the purpose of creating student-specific learning paths and have produced positive outcomes in Computer Science and Engineering such as improved student engagement and better student understanding of material. Moreover, a platform built at the University of Ruse [6] provides doctoral students with personalized learning paths supplemented with personal reporting to facilitate autonomous intellectual conclusions.

Shifting specifically to lesson creation and management features, existing learning management systems like Moodle

[4]Accreditation Board for Engineering and Technology, Inc. is a nonprofit organization that handles university accrediation in the domains of applied and natural science, engineering, computing and technology

and Brightspace [7], while heavyweight in comparison to our current platform, provide similar mechanisms for managing online lessons. However, what these systems definitely have in scope, third-party integrations and global reach, they lack in a simplicity and an allowance for creative freedom. Moodle documentation, for example, states that in order to create a lesson, it is required that you have a clear idea of what your lesson should look like before you start, and it even recommends having a notepad on hand to keep track of the lesson's complexity [8]. Moreover, neither Brightspace [9] nor Moodle offer a simple component-drag-and-drop interface for creating nesting-based personalizable lessons.

V. Conclusion & Future Work

The platform described in this paper is a freely available e-learning solution to be used not only as a supplemental teaching tool for traditional instructor-student learning, but as a teaching tool for anyone who wishes to teach, and as a learning tool for anyone who wishes to learn. From the platform itself, "We invite you to teach others what you know - whether that's age old fettuccine recipes or the ins and outs of astronomy, or something in between, is completely up to you." The primary purpose of this platform manifests in its simple, nesting-based mechanism for developing highly personalized and engaging lessons. The depth of nesting, or the **degree of personalization**, applied to a lesson is determined fully by the lesson creator. In return, the platform transfers control over the lesson's path to the student, facilitating varying degrees of autonomous learning which has been shown to be highly positively correlated with life-long learning[5] [10].

An easy-to-use toolkit of different component types allows one to take a creative, LEGO-like approach to building lessons; this aligns with [11] where it is discussed that utilizing creative teaching strategies has a profound positive effect on student learning. The available toolkit makes it possible to teach students by engaging them in what is more of a choice-driven game than a lecture, which channels the benefit of gamified education discussed in [12].

The platform's Reporting dashboard provides instructors with a window through which they can view statistics about their collective lesson set, their individual lessons, and each lesson's individual components. The dashboard provides clear insight regarding what students are understanding and what they are not, as well as well as insight about the instructor's own teaching strategies down to whether or not specific explanations are sufficient for clarifying muddy waters.

Our plans for future development on the platform are geared primarily toward lesson creation and planning. With our current system, all lessons created are independent from each other and there is no way to establish any order or dependency among them apart from directing students in the lessons themselves to finish prerequisite lessons beforehand. Of course, if fundamentals are skipped, students are less likely to do well on more complex material, resulting in less-than-ideal lesson reports. We aim to integrate a higher-level organization structure for grouping lessons into **lesson series** accompanied by the option to require ordered lesson viewing to account for these dependencies. We also aim to create a collaborative lesson building mechanism for instructors that wish to work together on creating lessons without having to share account details.

[5]Lifelong learning is voiced in a person's self-motivated an ongoing pursuit of knowledge

References

[1] A. Ziv, "Teaching and learning with humor," *The Journal of Experimental Education*, vol. 57, no. 1, pp. 4–15, 1988. [Online]. Available: https://doi.org/10.1080/00220973.1988.10806492

[2] S. Sasidhar, A. Muruganantham, W. Soh, Y. J. Soo, C. Tan, and R. Suppiah, "Augmenting experiential learning with a game-based student response system," in *2018 IEEE Frontiers in Education Conference (FIE)*, 2018, pp. 1–6.

[3] I. Cabezas, "On combining gamification theory and abet criteria for teaching and learning engineering," in *2015 IEEE Frontiers in Education Conference (FIE)*, 2015, pp. 1–9.

[4] Y. Deng, D. Lu, C. Chung, D. Huang, and Z. Zeng, "Personalized learning in a virtual hands-on lab platform for computer science education," in *2018 IEEE Frontiers in Education Conference (FIE)*, 2018, pp. 1–8.

[5] Q. M. Clark and J. V. Clark, "Personalized learning tool for thermodynamics," in *2018 IEEE Frontiers in Education Conference (FIE)*, 2018, pp. 1–5.

[6] P. Zlatarov, G. Ivanova, and D. Baeva, "A web-based system for personalized learning path tracking of doctoral students," in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2019, pp. 773–778.

[7] Y. Kats, *Learning Management System Technologies and Software Solutions for Online Teaching: Tools and Applications: Tools and Applications*, ser. IGI Global research collection. Information Science Reference, 2010. [Online]. Available: https://books.google.com/books?id=2OsPGIvgjGwC

[8] Moodle. (2019, Oct.) Building lessons. [Online]. Available: https://docs.moodle.org/38/en/Building_Lesson#Planning_your_lesson

[9] J. S. D2L. (2017, Jul) Demo of brightspace lessons. YouTube. [Online]. Available: https://www.youtube.com/watch?v=L6oogmIa8K8

[10] C. YURDAKUL, "An investigation of the relationship between autonomous learning and lifelong learning," *International Journal of Educational Research Review*, pp. 15 – 20, 2017.

[11] J. L. Rinkevich, "Creative teaching: Why it matters and where to begin," *The Clearing House: A Journal of Educational Strategies, Issues and Ideas*, vol. 84, no. 5, pp. 219–223, 2011. [Online]. Available: https://doi.org/10.1080/00098655.2011.575416

[12] J. Hamari, D. J. Shernoff, E. Rowe, B. Coller, J. Asbell-Clarke, and T. Edwards, "Challenging games help students learn: An empirical study on engagement, flow and immersion in game-based learning," *Computers in Human Behavior*, vol. 54, pp. 170 – 179, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S074756321530056X