INFORMATION SYSTEMS EDUCATION JOURNAL

Volume 18, No. 6
December 2020
ISSN: 1545-679X

In this issue:

iscap
Information Systems & Computing Academic Professionals

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **ISCAP** (Information Systems and Computing Academic Professionals). Publishing frequency is six times per year. The first year of publication was 2003.

ISEDJ is published online (http://isedj.org). Our sister publication, the Proceedings of EDSIGCON (http://www.edsigcon.org) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the EDSIGCON conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is under 40%.

Information Systems Education Journal is pleased to be listed in the Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org. Special thanks to members of EDSIG who perform the editorial and review processes for ISEDJ.

INFORMATION SYSTEMS EDUCATION JOURNAL

The impact of an interactive textbook in a beginning programming course

Joni K. Adkins
jadkins@nwmissouri.edu

Diana R. Linville
dianar@nwmissouri.edu

Charles Badami
cbadami@nwmissouri.edu

School of Computer Science and Information Systems
Northwest Missouri State University
Maryville, MO 64468, USA

Abstract

Online textbooks allow instructors to provide interactive and engaging activities for students. In this paper, we look at how providing an interactive online textbook is utilized and valued in a beginning computer programming course. In addition, we compare the utilization of the online textbook to the student final course grade. Our findings suggest that students would rather use an online textbook and the level of engagement in the online textbook activities was positively related to a student's final course grade. These findings encourage us to continue evolving and improving the interactive features provided in the online textbook.

Keywords: online textbooks, interactive textbooks, active learning, computer programming

1. INTRODUCTION

Advances in technology afford new ways for students to learn. For example, today's students are more comfortable using online sources and the availability of free learning resources such as Codeacademy and Khan Academy have changed the education landscape. Educators looking for ways to improve student learning and engagement have developed online resources, including online textbooks to help students learn computer programming.

The hope is that an interactive online textbook may be more appealing to students, thus increasing their use of the resource. Current research shows that many students do not read textbooks as assigned. Reasons include poor study habits, lack of motivation, and poor time management (Starcher & Proffitt, 2011). Some students do not even have the textbook due to the high price (Robinson, 2010). Brost and Bradley (2006) found that students may not read the textbook because they know the teacher will cover the material in class anyway.

We sought to answer three research questions in this study. One, what classroom activities and assignments do the students view as valuable? Two, how do the students perceive the usefulness of the online textbook readings, activities, and quizzes? Three, is a student's online textbook grade indicating their participation and effort in the online textbook a valid predictor for the overall course grade?

This paper begins with a literature review related to interactive textbooks. Then the development

of our online tool is discussed along with the format of the course and implementation details of the new tool. Results from student surveys and data analysis to answer the research questions are then shared.

2. LITERATURE REVIEW

The pedagogical rationale for this study was based on active learning defined as activities that encourage students to engage with course materials and increase critical thinking (Lumpkin, Achen, & Dodd, 2015). Many studies have found that students like active learning as well as discovering that students can retain content better (Hyun, Ediger, & Lee, 2017). The use of an interactive textbook requires students to be actively involved in their learning experience. The majority of the literature surrounding interactive, online textbook resources in computer science education from the last ten years seems to be concerned with the analysis of student improvement in related courses. Other studies have focused on student perceptions and usage of online textbooks, and some have centered on effective design components of such a resource.

The research that evaluates student improvement when using an interactive resource varies in sample size and thoroughness, but much of it seems to agree in finding positive correlations. Aldubaisi (2014) examined computer science student performance in conjunction with the use of an interactive e-textbook, one apparently developed for the study by the author. Although this was a short-term study, it resulted in a positive reaction and better performance from the students who participated. Edgcomb et al. (2015) embarked on a long-term, thorough study across three universities, four programming classes, and almost 2,000 students for multiple terms (same instructors). They found significant statistical improvement in both exam scores and final letter grades, when comparing users of an interactive text versus a static one. A pilot study by Farnqvist, Heintz, Lambrix, Mannila, and Wang (2016) involved an online tool called OpenDSA, used for data structures and algorithms courses. Their main finding was that students scored better on the final exam, while also showing a preference for the online tool in log data and questionnaires. A study by Alshammari and Pivkina (2017) compared discrete math and programming courses, in terms of early versus late completion of interactive reading assignments and student performance. Notably, they found that early finishers of interactive reading did better in discrete math, but there was no significant improvement for the analogous programming students; however, the authors concluded that another factor may have to do with how essential the assigned reading is to the course in question.

Studies that mainly investigate student perceptions of interactive textbook material seemed to concur that feedback is generally positive and usage is increasing. Warner, Doorenbos, Miller, and Guo (2015) did a quantitative study of an interactive, online computer programming text using data gathered from over 43,000 users. They found that all types of students (high school, college, online only) used the interactive components extensively, and many used the resource by jumping around, rather than just sequentially. Research by Pollari-Malmi, Guerra, Brusilovsky, Malmi, and Sirkia (2017) focusing on a Python course in Finland found that there was better student motivation, learning, and feedback regarding interactive texts versus static texts. The authors noted that other differences in teaching methods could have also contributed to the results, but any effect was deemed to be small. In addition, there was a flipped classroom study by Davenport (2018) that involved computer programming tutorials in a meteorology course. Although earlier studies suggested negative perceptions of this flipped methodology (including the interactive resources), especially toward the end of the semester, this particular study related to computer programming found that the majority of students recognized the benefits.

Finally, the design studies each offered suggestions for effective interactive components, but from different perspectives. The resource presented by Way (2016) was an interactive Java programming text and was presented in a self-justified manner. Notable design elements advocated by the author included active links to content, interactive coding, animations, and quiz-like checkpoints. In contrast, Ericson, Roger, Parker, Morrison, and Guzdial (2016) offered a well-tested and developed design study, built upon previous studies by the same authors, which included different iterations of the interactive text, as well as teacher and student observations and experiments. The major design recommendations proposed included combining worked examples, practice, and exercises at the end of chapters. Given the interest in studying interactive textbooks and

their positive impact on students, we decided to explore creating our own interactive resource.

3. COURSE DEVELOPMENT & DELIVERY

At this institution, the first foundational programming classes are taught in a sequence of three courses: Computer Programming I, Computer Programming II, and Data Structures. The Computer Programming I course is an introductory course currently taught in Python that covers basic programming concepts including types and operators, control structures, files, functions, and classes. A committee of faculty in the School of Computer Science and Information Systems (CS/IS) determine these topics.

In previous semesters, the course content was delivered using PowerPoint notes, text-based exercises, and projects that were provided through the course management system. The instructors utilized the PowerPoint notes to cover the programming concepts. The text-based exercises and projects were then completed by the students and submitted for grading. In addition, students were provided a printed textbook as a secondary resource.

In this course format, the provided printed textbook was not required to be utilized by students because it was not integrated into the course materials. Students could utilize it to read additional information on a topic or see other code examples, but there were no assigned readings or assignments from it. The main reason for this was that the textbook contained more information and topics than what was covered in the course. In addition, the concepts were introduced in a different sequence from the order in which the course was organized. The instructors determined that they wanted to provide the students with a textbook that covered only the topics the course introduced and in the sequence in which they were covered. At the same time, they wanted to create engaging components that would enrich the content and give students opportunities to practice the concepts. These factors motivated the instructors' desire to create an online interactive textbook that would do the following: 1. Incorporate the topics in the sequence introduced for this course. 2. Provide students immediate feedback when practicing basic programming concepts to help prepare for quizzes and exams. 3. Give students with different learning styles and/or disabilities access to online assistive technologies.

The online textbook was created in three phases. The first phase was to create the content. Following the outline and sequence of topics previously used in the course, the instructors divided the topics into seven chapters. Chapters were then separated into pages. Each page was then constructed into numbered sections that covered subgroups of the chapter topics. An example of a chapter outline follows.

Chapter 1
- Page 1
  - I. Intro
  - II. Output/Comments
  - III. Identifiers/Data Types
- Page 2
  - IV. Numeric Data
  - V. Input
  - VI. Turtle Graphics

The sections included interactive activities, which allowed students to check their understanding of the content covered in that section and receive immediate feedback. These activities ranged in format from multiple choice, fill in the blank, and matching questions. The sections also included what the instructors called an interactive code writer, which is an Integrated Development Environment (IDE) that has predefined code examples in it. Students can run, modify, and write code directly in the code writer. This gave them the ability to observe how the code executes and to view how modifications to the code affected the output. At the end of each page, a quiz was available for the students to test their knowledge of the topics covered.

During the next phase, the publisher and the instructors worked together to review all content and test all interactive units to make sure they functioned correctly. A small scale usability test was then conducted with a student who had previously taken the course. They provided feedback as to navigation and ease of use of the online textbook. The last phase was implementation of the online textbook during the spring 2019 semester. All sections of the course offered during that semester utilized the online textbook. There was not a control group because our school requires the use of the same textbook for all sections of a course.

The instructors introduced the online textbook the first day of class. Students then set up their account within the online textbook using the access code given to them by the instructor. Instructors familiarized students with the navigation of the online textbook and how to

work through the interactive components. Students were expected to work through the content in the online textbook prior to class and to practice the concepts. Students were instructed that none of the activities would be graded but were encouraged to use the content and activities to help them prepare for class, quizzes, and exams.

During class, instructors created code examples in Thonny, the IDE used in this class, and traced examples on the board to reinforce the concepts the students completed in the online textbook. Most class periods began with a short daily practice problem. Outside of class, students worked on longer module programming projects, worksheets, and short coding problems in a discussion format.

4. DATA ANALYSIS

The instructors gave an optional anonymous survey at midterm to gather data regarding the online textbook, course assignments, and in-class activities. The students were also asked open-ended questions regarding what they liked and did not like about both the online textbook and the class in general. Giving the first survey at midterm allowed the instructors to address concerns and make adjustments during the semester. Forty students took the midterm survey. The majority of the students who enroll in the course are freshman computer science majors but other majors also take the class including GIS, math, and digital media.

	Instructor 1 (n = 25)		Instructor 2 (n = 15)		df = 38	
	M	*SD*	*M*	*SD*	*t*	*p*
Project	4.12	1.2	4.27	.70	-.43	.670
Wkshts	3.96	1.0	3.80	1.0	.48	.633
Discuss	3.88	.88	3.87	1.1	.04	.966
Quiz	3.88	1.0	3.53	1.5	.82	.420
Videos	3.24	1.8	3.33	1.7	-.16	,873
Daily prac.	4.16	.85	4.20	1.1	-.13	.897
Thonny demos	4.68	.63	4.67	.62	.07	.948
Tracing	4.48	.77	4.13	.83	1.34	.189

Table 1: Differences between instructors

To answer the first research question, the students were asked at midterm to assign a score (1-5 with 5 being the best) to each class component. There was a choice "have not used" to select if they had not used that component. Data analysis was done to see if there were any differences in the class components between the two instructors. The independent samples t-tests indicated no significant differences between instructors so the students were combined into one sample for the remaining tests. Table 1 shows the results of these t-tests.

Table 2 shows the mean midterm scores for each component as rated by the students. Overall, the scores were positive with higher numbers associated with the activities that were done during class time and the programming projects done mostly outside of class. All components of the course, the online textbook, assessments, projects, discussions, and worksheets were closely related and covered the same material in different ways. This was possible since the course instructors wrote the online textbook.

Class Component	Mean 1-5 scale n = 40
Module programming projects	4.18
Worksheets	3.90
Discussion coding problems	3.88
Quizzes given in class	3.75
Lightboard tracing videos	3.28
Daily practice problems	4.18
Class demonstrations in Thonny	4.68
Tracing on whiteboard in class	4.35

Table 2: Mean scores of class components at midterm

Students had the option to share comments about what they liked about class and what they would like to have changed. The answers to these questions were analyzed to determine the most frequent comments. The most prevalent remark was to continue with the coding examples in Thonny. Since the instructors were no longer using PowerPoint lectures to cover the material, they often went into the IDE and typed Python code and comments and had the students follow along with them. The next two most frequent comments were to keep doing the module programming projects and the daily practice. The module programming projects were larger assignments that were completed mostly outside of class while the daily practice worksheets were like what many call bell work as they were handed out at the beginning of the class and the students were given the first 5-10 minutes to complete the worksheet which required them to predict code output or write code. The teachers would then review the daily practice before continuing class, and the

students got to keep the sheet. These were not graded. The comments about what to change included "more examples in Thonny" and "more complex in-class assignments." Clearly going through code in the IDE in class was viewed as valuable to students.

The students were also asked at midterm how much time they spent with the online textbook each week. We did not have the students keep a reading log so it was a student-provided estimate. Table 3 shows the breakdown of their answers with 53 percent of the students reporting they spent 1-2 hour each week using the online textbook.

Response	Number of students (n = 40)
Do not use the online textbook	4
Less than 1 hour	12
1-2 hours	21
2 or more hours	3

Table 3: Weekly hours with online textbook

The students were also asked to score the online textbook components on a five-point scale with 5 being "very good" and 1 being "very poor." There was an option "have not used" so students who did not use that component would not judge it. The components were: readings, interactive activities, quick quizzes, and the interactive code writer. At the end of the course, the students were asked the same questions about the textbook. Thirty students answered the second survey.

	Midterm mean (n = 40)		Final mean (n = 30)		df = 68	
	M	*SD*	*M*	*SD*	*t*	*p*
Reading	3.75	1.4	3.73	1.3	.50	.652
Activity	3.43	1.6	3.40	1.7	.06	.407
Quizzes	2.90	1.9	2.97	2.2	-.1	.445
Code Writer	2.90	1.8	2.67	2.0	.52	.063

Table 4: T-test results comparing midterm and final evaluation of online textbook

After the midterm evaluation, the instructors realized that some students were not using the online textbook so the next lesson was taught in class with the online textbook. We wanted to know if the exposure in class changed their attitudes toward the book so students were asked questions about the online textbook at the end of the course. Independent samples t-tests were done to see if there were significant differences in the responses between the midterm and the final survey. Table 4 shows the results. There were no significant differences in how students rated the online text components between the midterm and final evaluations.

The component of the online textbook that was rated highest was the readings. Qualitative comments also reflected that the way the online textbook was written was well liked. There were several positive comments that the online text was "easy to read," "short, and "all information was there." Another popular theme regarding the online textbook was the interactive part. Students repeatedly mentioned that they liked the built-in quizzes and activities and liked to be able to work on their own and get feedback immediately.

Students also realized some challenges when working with the online textbook. The most common comment dealt with some kind of a technical issue where there were errors or a refresh was required to get the book to work. Some students mentioned they would like to have immediate feedback on the correct quiz questions. The feedback was available but students had to go to the online gradebooks to see which ones they missed. If they were just wanting to see the answers without taking the quiz first, then that option was not available. In addition, a few mentioned there were some navigation and search issues that made it hard to use. Others mentioned that the navigation and search capability was a positive.

The day that the instructors demonstrated the online textbook, the interactive code writer did not work as expected so the students were reminded they could always copy and paste the code into Thonny to test if the code writer did not work. In the final evaluation, the students were asked for their preference for using Thonny or the interactive code writer. Over 83 percent of the students said they would rather copy and paste code from the online textbook into Thonny instead of using the included interactive code writer. This is valuable feedback for future direction of the online textbook. Incorporating the interactive code writer was a challenging part of developing the textbook and required additional cost. Removing that component and having students copy and paste code into their preferred IDE may be a better fit for the book. We will need to explore this in order to keep the interactive component viable. Students were also asked about their preference for an online or paper textbook. The results were overwhelmingly in favor of having an online

textbook instead of a paper textbook with 87 percent preferring online.

In addition to the data from the student surveys, the instructors were also able to obtain data regarding the use of the online textbook through the publisher's gradebook. The activities and quizzes from the online gradebook were not included as part of the course grade. However, instructors could see the online gradebook to tell which students had completed the activities and quizzes. For the activities, students received a 1 if they submitted the activity and a 0 if they did not. By submitting, they would learn if they got the answers correct. They received a 1 if they submitted, regardless of the accuracy of their work. There were a total of 47 activities in the online textbook. There were 15 quick quizzes in the online textbook. Students were timed on the quizzes but could take them multiple times, and the highest score was recorded in the online textbook gradebook. The quizzes were each worth 10 points. The total points available was 197 with 47 from activities and 150 from quizzes. Of the 36 students completing the course, 11 students (31 percent) showed no or very low interaction with the online textbook, earning fewer than 10 points in the online gradebook. Measuring the time spent reading or the amount of reading done in the textbook was not available through the online gradebook so could not be included in this analysis. All students who completed the course and received a grade were used in this analysis.

Final course grades are approximately 70 percent quizzes and exams, 15 percent programming projects, and 15 percent discussions and worksheets. To answer our third research question, we used regression to discover whether the grade from their online textbook gradebook was a valid predictor for their overall class percentage. Other predictors that were tested were the average quiz score, the total quiz score, the total number of activities completed, and the total number of activities and quizzes completed. A correlation matrix was generated and as expected, Pearson's coefficients ranged from .807 to .984, indicating a high level of correlation between the independent variables. Since multicollinearity existed as the predictors (independent variables) were related, each of these predictors was tested in simple regression (Hair, Black, Babin, Anderson, & Tatham, 2006). The best predictor for the overall course grade was the online textbook grade participation score. A significant regression equation was found ($F(1, 34) = 9.99$, $p < .003$), with an R^2 of .227. Participants' predicted course grade is equal to 77.34 (out of 100) plus .097 points for each point increase in online textbook participation. Table 5 shows the results of this analysis.

The average quiz score, total number of quizzes taken, total number of activities completed, and the total number of both activities and quizzes were all significant predictors as well but were not better than the online gradebook participation score.

	F (1,34)	R^2	*p*	b_0	b_1
Online text grade	9.9	.227	.003*	77.34	.097
Avg quiz score	8.9	.161	.015*	77.37	1.12
Total quiz taken	6.7	.164	.014*	78.66	.846
Total activ. comp.	9.3	.214	.004*	75.73	.309
Total quiz & activ. comp.	9.2	.214	.005*	76.17	.240

*Significant

Table 5: Simple regression results with course grade as dependent variable

5. DISCUSSION OF FINDINGS

The publisher's online gradebook score as a predictor shows that the effort that students put into the both activities and the quizzes when using the online textbook were relevant. This finding indicates completing activities as well as trying to do well on the quizzes (versus just attempting them) are better predictors of a student's final course grade over just viewing the activities. This finding reinforces that student interaction with online materials can lead to learning gain as also shown in Pollari-Malmi et al. (2017) and Farnqvist et al. (2016).

The researchers were encouraged with the positive feedback regarding the use of an online textbook. This finding contradicts what Robinson (2010) found in her study regarding preference as the majority of the students purchased a paper copy. This is likely due to increased acceptance in online materials in the last decade. Pollari-Malmi et al. (2017) also found increased usage in e-textbooks over pdf's. The textbook used in this study was offered free to all students, but only 69 percent used the book, reinforcing Robinson's (2010) finding that many

students do not use a textbook even when provided free of charge.

Students scored reading the online textbook as the highest component. As previously mentioned, the online gradebook does not measure the amount of time that students spend reading so it's hard to know whether reading had a confounding effect on the results. Future studies will need to seek a better way to measure reading to determine its role in the final course grade.

Student comments regarding the ability to search, find, and navigate the online textbook were mixed. This could be due to some students using the book more to learn the features or there could be some usability issues that could be addressed. We will review the navigation and search and add some brief instruction in class so students will know how to use the online textbook. In addition, students may or may not have known how to find quiz feedback so that will also be part of our instructions in the future.

There are limitations to generalizing the results of this study. A larger sample size would make the results stronger. In addition, the dependent variable was course grade, and many factors influence final course grade other than the use of the online textbook. Continuing this study into future semesters will allow us to learn more about the impact of this online textbook.

6. CONCLUSION

The overall goal of this study was to examine the degree of utility and value of using an interactive online textbook in a computer programming course. Through analysis of surveys and data collected during a full term of using this resource in multiple sections of a beginning programming course, we have endeavored to answer three questions: what classroom activities were viewed as valuable by the student; how do students perceive the online textbook's usefulness in terms of its activities; is a student's online textbook grade a valid predictor for the overall course grade? Our findings were encouraging in that students were mostly positive in their feedback about the textbook, and that valuable information about the effectiveness of various classroom activities was collected. Additionally, we have data connecting the use of the online resource to a student's performance.

Educational techniques and student populations evolve constantly, which makes iterations of research in this area continually necessary. This particular topic is no different. As interactive online resources become more sophisticated and ubiquitous, no doubt there will be many opportunities for future research on this subject and improvement of these tools.

7. REFERENCES

Aldubaisi, F. (2014). *E-textbook evaluation study.* Auburn University.

Alshammari, K., & Pivkina, I. V. (2017). Relationship between time management in courses with online interactive textbooks and students' performance. *2017 IEEE Frontiers in Education Conference*, 1-5.

Brost, B. D., & Bradley, K. A. (2006). Student Compliance with Assigned Reading: A Case Study. *Journal of Scholarship of Teaching and Learning, 6*(2), 101-111.

Davenport, C. E. (2018). Evolution in student perceptions of a flipped classroom in a computer programming course. *Journal of College Science Teaching, 47*(4), 30-35.

Edgcomb, A. D., Vahid, F., Lysecky, R., Knoesen, A., Amirtharajah, R., & Dorf, M. L. (2015). Student performance improvement using interactive textbooks: A three-university cross-semester analysis. *Computer Science and Engineering*, 1-17.

Ericson, B., Roger, K., Parker, M., Morrison, B., & Guzdial, M. (2016). *Identifying design principles for CS teacher ebooks through design-based research.* Paper presented at the ACM International Computing Education Research, Melbourne, Australia.

Farnqvist, T., Heintz, F., Lambrix, P., Mannila, L., & Wang, C. (2016). *Supporting active learning by introducing an interactive teaching tool in a data structures and algorithms course.* Paper presented at the SIGCSE, Memphis, TN.

Hair, J. F., Black, W. C., Babin, B. J., Anderson, R. E., & Tatham, R. L. (2006). *Multivariate Data Analysis* (Sixth ed.). Upper Saddle River, NJ: Pearson.

Hyun, J., Ediger, R., & Lee, D. (2017). Students' satisfaction on their learning process in active learning and traditional classrooms. *International Journal of Teaching and Learning in Higher Education, 29*(1), 108-118.

Lumpkin, A. L., Achen, R. M., & Dodd, R. K. (2015). Student perceptions of active learning. *College Student Journal, 49*(1), 121-133.

Pollari-Malmi, K., Guerra, J., Brusilovsky, P., Malmi, L., & Sirkia, T. (2017). On the value of using an interactive electronic textbook in an introductory programming course. *17th Koli Calling International Conference on Computing Education Research*, 168-172.

Robinson, S. (2010). Student use of an online textbook: Even if it's free, will they buy it? *Allied Academies International Conference: Proceedings of the Academy of Educational Leadership (AEL), 15*(1), 44-49.

Starcher, K., & Proffitt, D. (2011). Encouraging students to read: What professors are (and aren't) doing about it. *International Journal of Teaching & Learning in Higher Education, 23*(3), 396-407.

Warner, J., Doorenbos, J., Miller, B. N., & Guo, P. J. (2015). *How high school, college, and online students differentially engage with an interactive digital textbook.* Paper presented at the 8th International Conference on Educational Data Mining.

Way, T. (2016). An improved approach for interactive ebooks. *Annual Joint Conference Integrating Technology into Computer Science Education*, 248-249.