Sánchez, J. (2016) Editor. Nuevas Ideas en Informática Educativa, Volumen 12, p. 704 - 709. Santiago de Chile.

I3S Simulator: An Open Educational Resource for Teaching Scheduling in Interactive Systems

Maria Lydia Fioravanti
University of São Paulo (ICMC – USP)
São Carlos, SP, Brazil
mlfioravanti@usp.br

Marcelo Koti Kamada
University of São Paulo (ICMC – USP)
São Carlos, SP, Brazil
marcelokkamada@gmail.com

Paulo Sergio Lopes de Souza
University of São Paulo (ICMC – USP)
São Carlos, SP, Brazil
pssouza@icmc.usp.br

Ellen Francine Barbosa
University of São Paulo (ICMC – USP)
São Carlos, SP, Brazil
francine@icmc.usp.br

ABSTRACT
Teaching and learning scheduling in interactive systems are challenging tasks. Many theoretical concepts may discourage learning due to their complexity and lack of practical applications. In an attempt to bridge such a gap, several resources, including Open Educational Resources (OERs), have been developed to improve learning and teaching in computing courses, including Operating Systems courses. This paper presents an open educational resource entitled I3S Simulator, which aims at facilitating the consolidation of knowledge by simulating the structure, functionality and performance of scheduling processes in interactive systems. It was analyzed by students and teachers and the results indicate their interest in adopting it as a supporting mechanism for the teaching and learning.

Categories and Subject Descriptors
• Applied computing~Interactive learning environments

General Terms
Algorithms, Human Factors, Theory.

Keywords
Open educational resource; Simulator; Scheduling in interactive systems.

1. INTRODUCTION
Teaching and learning are great challenges, especially at the university. Besides transmitting information, teachers must consider the different students' cognitive styles and learning strategies [1], since some students are visual learners, while others are auditory or kinesthetic learners, among other classifications proposed.

Another issue that may discourage graduate and undergraduate students is the disassociation between theory and practice [2]. The teaching of basic concepts and theory foundation with no link to their practical applications or no examples in the students' contexts is usual in computing courses.

In an attempt to bridge such a gap, several resources, including Open Educational Resources (OERs), have been developed to improve learning and teaching in computing courses. OERs have emerged as an important element of education, as they provide new perspectives for learning in the contemporary society, characterized by an intensive adoption of technologies and a more participatory and collaborative learning approach centered on the learners [3].

The concept of Open Educational Resources can be interpreted from more restrictive to broader definitions. The most common definition refers to OERs as teaching, learning and research materials that reside in the public domain or have been released under an intellectual property license that allows their free use [4].

This paper presents an Open Educational Resource entitled I3S Simulator, whose objective is to assist the teaching and learning of introductory concepts of scheduling processes in interactive systems through an Operating Systems class. Concepts related to operating systems are not trivial, especially when their understanding requires students' abstraction. In this sense, I3S aims at helping the whole process of teaching and learning of Operating Systems. The idea is not to replace the teacher, but to use the OER as a support for both teaching and extra-class activities performed by students. Through visual interaction, students can understand the subject better, without being discouraged by the traditional teaching method.

The paper is structured as follows: Section 2 presents the traditional paradigm of teaching scheduling in interactive systems; Section 3 briefly describes the OER proposed; Section 4 discusses the survey conducted for the evaluation of the I3S Simulator; Section 5 addresses the related work; finally, conclusions and perspectives for future work are provided in Section 6.

2. TRADITIONAL TEACHING OF SCHEDULING IN INTERACTIVE SYSTEMS
When a computer is multiprogrammed, it frequently has multiple processes competing for the CPU at the same time. This situation occurs whenever two or more processes are simultaneously in the ready state. If only one CPU is available, a choice has to be made which process to run next. The part of the operating system that makes the choice is called the scheduler and the algorithm it uses is called the scheduling algorithm [5].

The problem of scheduling the resources of a computer is decades old. While the first machines were simple number crunchers capable of running one program at a time, it soon became apparent that there is a need to allow for several programs to be served concurrently.

According to Tanenbaum [5], in different environments different scheduling algorithms are needed. This situation arises because different application areas (and different kinds of operating systems) have different goals. In other words, what the scheduler should optimize for is not the same in all systems. Three environments worth distinguishing are: batch, interactive and real-time. In this work, we will consider only interactive.

Scheduling in interactive systems has been traditionally taught through lectures in Operating Systems courses. Blackboards and slide presentations are used as a support to the explanations. However, these are very static methods, since the teacher will follow the class plan rigorously, with little room for changes.

Our previous experience shows that the concepts of scheduling in interactive systems are taught in the context of Process Scheduling, which requires a two-hour class. During the lecture, the teacher focuses on the broader topics related to Process Scheduling, and little time is available for particularities of the scheduling of interactive systems. Moreover, the available resources do not encourage teachers to approach such particularities.

The traditional approach for teaching scheduling in interactive systems has positive and negative aspects. Regarding positive aspects, we can point out:

- Content is broader and the available literature reports many details;
- Classes highlight structural aspects of the subject;
- Students can read the textbook in advance and take more advantage of the classes (however, this is almost never done).

The following negative aspects can be highlighted:

- Students, in general, exhibit a passive behavior;
- Content requires a high level of abstraction to be understood;
- Students have problems with concentration and focusing;
- Didactic resources are static, which hinders interaction with students;
- Aspects related to functionality and performance are pointed out, but not properly exemplified.

Although slide presentations can be considered technology-enhanced teaching and learning, this teaching paradigm must be improved to be more dynamic, attractive and interactive.

3. AN OVERVIEW OF I3S SIMULATOR

In an attempt to bridge the gaps addressed, we have adopted an approach that includes the use of an OER called I3S Simulator, available at http://lasdpc.icmc.usp.br/~ssc640/pos/i3s/ as a supporting mechanism for teaching scheduling in interactive systems during classes.

I3S is an acronym for Interactive Systems Scheduling Simulator, i.e., it is a simulator that assists in the teaching and learning of introductory concepts of scheduling processes in interactive systems, specifically highlighting the structural and functional aspects of scheduling algorithms [5], namely (i) round-robin, (ii) priority, (iii) multiple queues, (iv) shortest job first (SJF), and (v) lottery. The performance of the five algorithms can also be compared to the same set of processes, which enables the student to visualize the advantages and limitations of each algorithm.

For the development of the simulator, we used an MVC architectural pattern to decouple business, presentation and control logic. The Model consists of scheduling algorithms (business logic), implemented in Python programming language. The View was implemented by the Bootstrap framework for GUI with Javascript scripts, handling all user's interactions. The Controller, developed in PHP, provides an interface with the options chosen by the user, executes them and sends the results to the View.

Since the simulator was developed as an OER, it should be self-contained, i.e., it should not require further material to be understood and used. Therefore, the OER should be intuitive enough, so that student can use it without supervision of a teacher/tutor. Figures 1 to 4 show all the steps to be taken by the student for running a simulation.

3.1 Step 1: Choice of the algorithm(s)

The student can choose one or more algorithms to run the simulation. Additionally, descriptions of each algorithm and its strengths and weaknesses are shown.

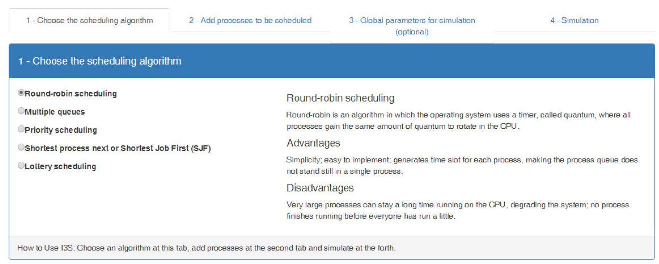


Figure 1. Step 1 - Choice of the algorithm(s)

3.2 Step 2: Addition of the processes

The student must add manually as many processes as he/she wishes to be scheduled for each algorithm previously selected. It can also be performed automatically via option “Use Suggested List”.

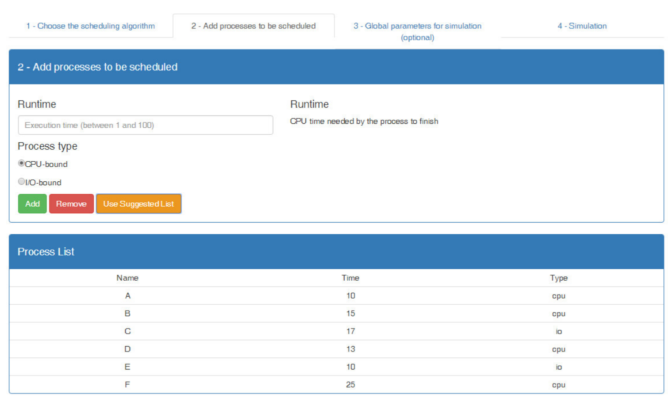


Figure 2. Step 2 - Addition of the processes

3.3 Step 3: Setup of global parameters

The student can optionally change some global parameters.

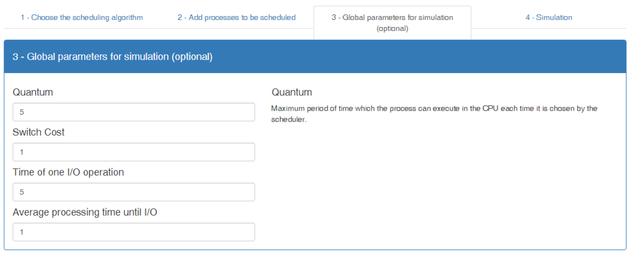


Figure 3. Step 3 - Setup of global parameters

3.4 Step 4: Running of the simulation

The student is able to see which scheduling algorithm(s) he/she has selected and must click on the button "Run simulation" to start the simulation.

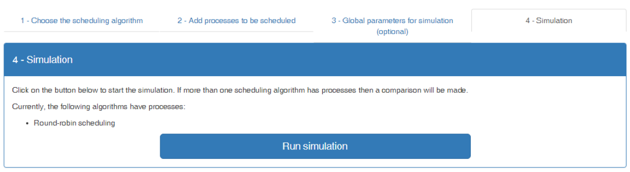


Figure 4. Step 4 - Running of the simulation

Explanations on the algorithms and each parameter are given, thereby, the student can learn or even establish the concepts.

After the simulation parameters have been set, the I3S simulator page is shown (Figure 5). The users can choose among running the simulation step by step, jumping to the last step, resetting it or even going to the homepage for a new simulation. If the simulation is run step by step, the simulator allows stepping forward and backwards and shows the user what occurs along with CPU usage, Total Executing Time and Context Switches to guide the user. The students interact more directly with the simulator and are motivated to adopt such a tool.

In its current version, I3S provides support for English and Portuguese languages through XML files containing translations. Actually, the system supports other languages with the addition of new translation files.

The use of I3S Simulator is predicted in two different scenarios, i.e., during classes and in extra-class activities.

In the first scenario, the OER helps the teacher to deliver the content in a more attractive way. The learners can visualize the scheduling effectively instead of imagining it, which increases their motivation. Furthermore, I3S enables the exemplification of aspects not considered in the traditional method, e.g. functionality and performance.

In the second scenario, the OER guides the self-education (i) as extra-class exercises in the context of an Operating Systems course and (ii) for learners interested in Operating Systems and who are not enrolled in any specific course.

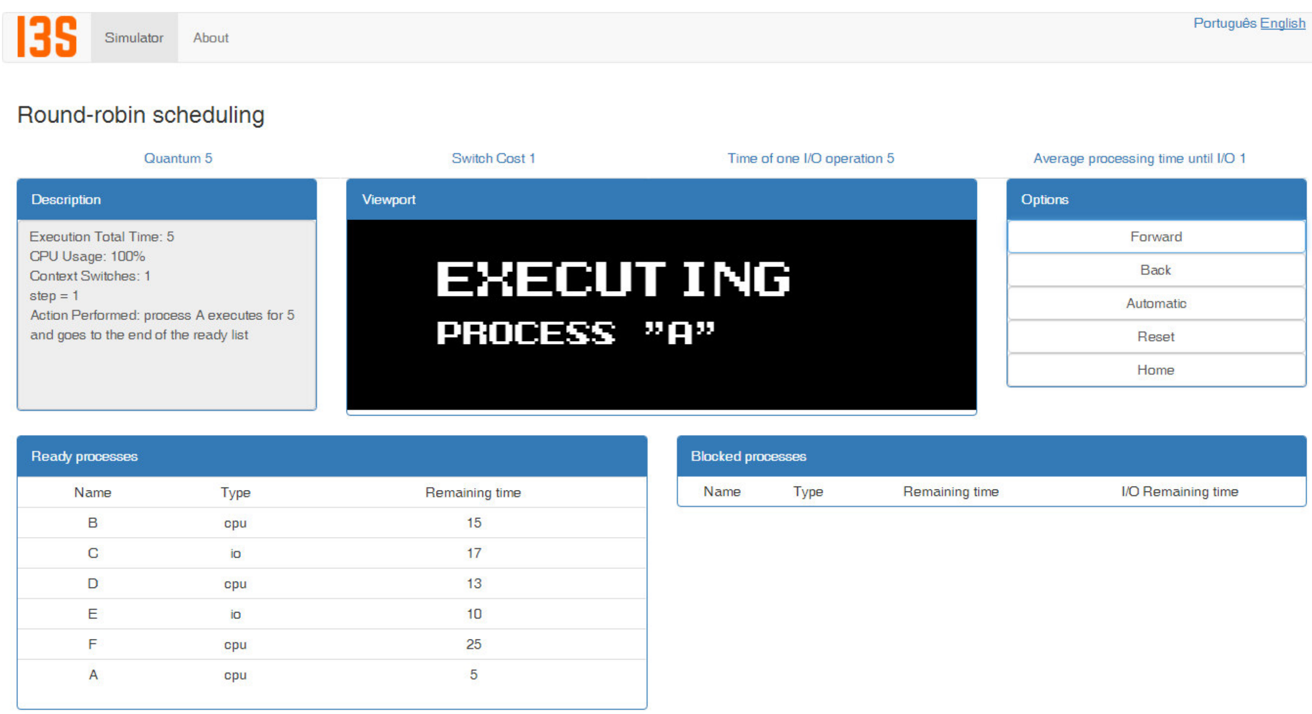


Figure 5. Simulator page

4. EVALUATION AND RESULTS

A survey on the opinion of students and teachers was carried out for a preliminary evaluation of I3S. The attitudes of undergraduate students, graduate students and teachers towards the adoption of I3S were considered.

The questionnaire applied consisted of 10 questions, shown in Table 1, and was sent via email to the evaluators.

Table 1. Survey Questions

Question	***Type***
Name	-
Evaluator degree	Multiple choice
OER relevance	Likert scale
OER adherence to its goals	Likert scale
OER user's interface	Likert scale
Use by the teacher in the classroom	Likert scale
Use by the student during extra classroom activities	Likert scale
Installation process	Likert scale
OER correctness	Open
Suggestions, strengths and weaknesses	Open

According to Figure 6, most answers were given by students and 6% of the evaluators were teachers.

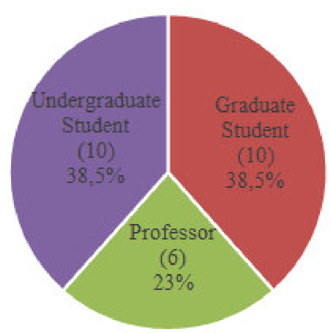


Figure 6. Evaluator degree

We asked about their attitude towards the adoption of the OER, regarding its relevance, adherence, user interface, usage and installation. To measure such items, we used likert scale with five options: Excellent, Great, Good, Regular and Poor. The only exception was the question about the installation, which contained the following options: Excellent; Great; Good; Regular/Poor; and Not applicable.

Figure 7 shows the evaluators' attitudes on the ease of installation. 92% of the evaluators found the I3S simulator easy to install (Excellent or Not applicable), since it requires no configurations and the user must only access the website. We observed the other 8% did not have the same impression and considered they were concerned about all the steps to deploy the tool in a server.

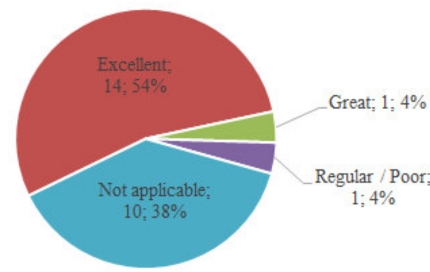
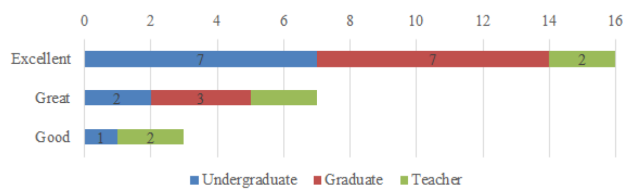


Figure 7. Installation process evaluation

Figure 8 shows the evaluators agreed on the OER relevance, although 11% chose "Good" option. The remaining 89% considered it "Great" or "Excellent". Two of the three evaluators that chose "Good" were teachers. We aimed at investigating the improvements that should be made, so that relevance could be increased from the teachers' point of view.

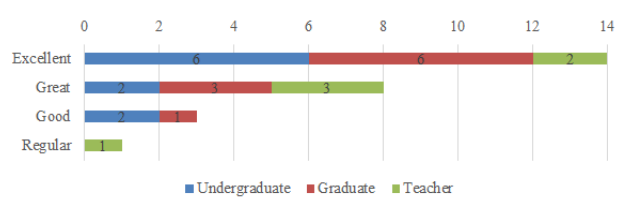


(a) Degree

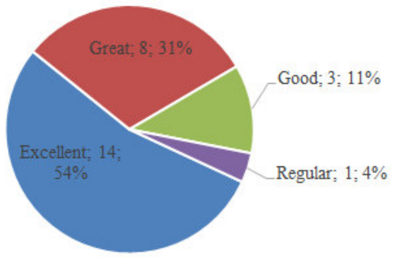
(b) Total

Figure 8. OER relevance

Figure 9 shows the results of OER adherence to its goals were similar to the OER relevance, except that one teacher answered it is "Regular". We understand it has fulfilled its requirements, since 85% of the evaluators answered "Excellent" or "Great".



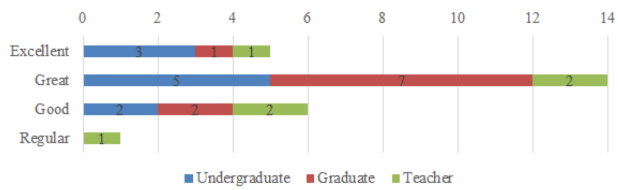
(a) Degree



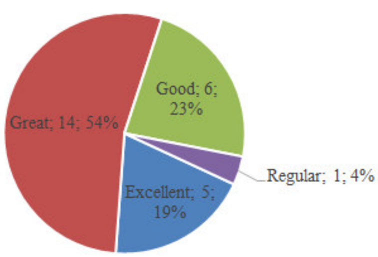
(b) Total

Figure 9. OER adherence to its goals

The graphic user's interface suited most evaluators, with 96% of positive evaluations, as shown in Figure 10. A teacher considered it “Regular”, but he/she did not leave any further comments.



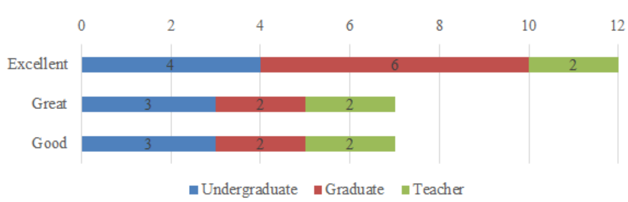
(a) Degree



(b) Total

Figure 10. OER User Interface

Another aspect evaluated was the “ease of use” by teachers and students. According to Figure 11, 46% of the teachers considered it “Excellent”. Their evaluations were balanced and “Excellent”, “Great” and “Good” categories had two votes each.

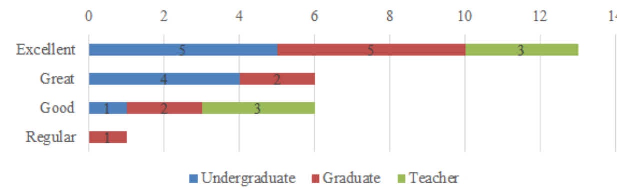


(a) Degree

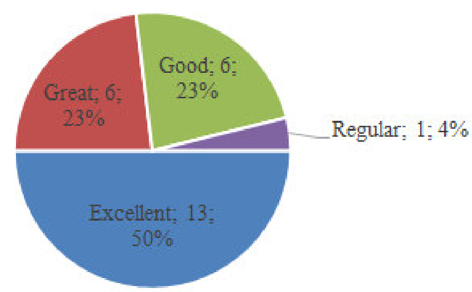
(b) Total

Figure 11. Use by the teacher in the classroom

Regarding the use by students, the scenario is different, as shown in Figure 12. 50% of the evaluators considered the OER “Excellent” to be used by the student alone, while 46% considered it “Great” or “Good” and 4% considered it “Regular”.



(a) Degree



(b) Total

Figure 12. Use by the teacher in the classroom

The 4% of “Regular” votes consisted of one vote by a graduate student who did not explain what could be improved to help in a self-guided study.

The last two questions were open and the evaluator could point out mistakes, suggestions, strengths and weaknesses of I3S. Only two of them pointed out mistakes, which are being investigated. The other answers consisted of positive comments and suggestions for improvements.

The preliminary results show I3S has been well-accepted and fulfilled the basic requirements of an OER.

5. RELATED WORK

We compared I3S to four other available simulators of different complexities and features that focus on process scheduling.

- **Process Scheduling Simulation Project [6]:** It is a VB.NET application that computes average turnaround time, waiting time and job throughput to evaluate the performance of an operating system.

- **CPU Scheduler Application [7]:** It is a Java application designed to simulate the short-term scheduler in an operating system.
- **Process Scheduling Simulator [8]:** It enables the user to experiment with various process scheduling algorithms on a collection of processes and compare such statistics as throughput and waiting time.
- **PSSAV [9]:** Process Scheduling Simulation, Analyzer, and Visualization (PSSAV) is an application that provides CPU process scheduling algorithm simulation.

As shown in Table 2, as the I3S Simulator is an open educational resource designed to be as complete as possible, it includes various scheduling algorithms of interactive systems, whereas the other simulators include only some of them and are more generic.

Table 2. Main features of the simulators

Features	[6]	[7]	[8]	[9]	I3S Simulator
Round-robin	X	X	X	X	X
Multiple queues					X
Priority		X		X	X
SJF	X	X	X	X	X
Lottery					X
GUI	X	X	X	X	X
Web			X		X
Tutorial	X	X	X	X	X
Source Download	X	X	X	X	X

Although the proposed simulators have a graphic user's interface to motivate their use by the students, only Process Scheduling Simulator and I3S Simulator are available on the web. The former is available via Java applet, while the latter runs on the web. Web availability facilitates the use of the tool by teachers and students because no installation process is required.

All simulators, including I3S, have tutorials to guide the users and an available source code.

6. CONCLUSIONS AND FUTURE WORK

This paper has presented an open educational resource, entitled I3S Simulator. We discussed its use as a supporting mechanism for teaching and learning in Operating Systems courses, focused on the teaching of scheduling in interactive systems.

We expect I3S will enable the abstraction of the complexity of the subject and students will better understand and fix the basic concepts by visual interaction, which facilitates knowledge consolidation.

In general, users were enthusiastic and positive about the use of I3S and its importance to the teaching and learning process. Although I3S still requires improvements, the available version has fulfilled the requirements for the teaching of basic concepts of scheduling in interactive systems.

As future work, we aim at a more complete evaluation of I3S for the measuring of the learning effectiveness. Moreover, the improvements suggested by the evaluators, such as animations and slight GUI changes will be considered and implemented in the OER. We can also adapt I3S to a mobile application, so that students can use it in their smartphones or tablets and reach a larger number of people. Finally, we highlight the need for the creation of initiatives to encourage teachers and students to use such a tool on a daily basis.

ACKNOWLEDGMENTS

The authors acknowledge Brazilian funding agencies (FAPESP, CAPES and CNPq) for their support.

REFERENCES

[1] R. Riding and S. Rayner, *Cognitive styles and learning strategies: Understanding style differences in learning and behavior*. Routledge, 2013.

[2] R. Matthews, H. S. Hin, and K. A. Choo, “Merits and Pitfalls of Programming Learning Objects: A Pilot Study,” in *Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia*, 2012, pp. 293–296.

[3] A. Kanwar, S. Uvalić-Trumbić, and N. Butcher, *A basic guide to open educational resources (OER)*. Vancouver: Commonwealth of Learning; Paris: UNESCO, 2011.

[4] William and Flora Hewlett Foundation, “Education Program: Strategic plan.” 2010.

[5] A. S. Tanenbaum, *Modern operating systems*. Pearson Education, 2009.

[6] A. Modi, Karim, and M. Yang, “Process Scheduling Simulation Project.” 2012.

[7] J. Weller, “CPU Scheduler Application.” 2006.

[8] S. Robbins and K. A. Robbins, “Empirical Exploration in Undergraduate Operating Systems,” in *The Proceedings of the Thirtieth SIGCSE Technical Symposium on Computer Science Education*, 1999, pp. 311–315.

[9] F. Adiputra and H. T. Hidayat, “PSSAV (Process Scheduling Simulation, Analyzer, and Visualization).” 2010.