Web-Based, Interactive Software for Engineering Economy Courses

Kevin D. Dahm and Ravi P. Ramachandran
Rowan University

David L. Silverstein
University of Kentucky

Abstract

A game simulating practical economic decision-making has been devised and integrated into a Rowan University course on engineering economics. Web-based software that administers the game was then developed with the support of an NSF-CCLI grant. During the fall 2003 semester, the software was used in a classroom setting for the first time as part of a course at the University of Kentucky.

The game itself challenges students to not only learn engineering economic principles such as present worth, but also to use them to make realistic economic decisions in a competitive setting. Each student starts with $10,000, and is presented with a list of investment opportunities. Students apply the principles learned in class to the possible investments and make decisions, such as how much to bid on a particular item in an auction. Additional investment opportunities are introduced throughout the semester. The required analysis grows in complexity as the students' knowledge base increases. The game is interactive; for example the owner of a factory must negotiate the price of raw materials he/she needs with the owner of a mine.

The software is web based, written using a combination of standard HTML and Active Server Pages with a Microsoft Access Database. Students use the software to manage their company, taking such actions as placing bids in an auction, borrowing money from a bank, purchasing raw materials, and setting production rates for mines and factories. Students can also use the software to access reports on the current status of their company- lists of assets, current bank balance, etc. Finally, students can propose transactions with other students using the software. Administration of the game in its current form requires very little effort on the instructor's part; he or she only needs to run one script approximately weekly to process student's actions.

This paper will describe the capabilities and functionality of the software in detail and discuss revisions resulting from the classroom test.

Introduction

Recent texts on engineering economics[1-4] all stress that the practical purpose of engineering economics is that it empowers the engineer to make sound investment decisions. This paper describes a game, integrated into an engineering economics course as a semester-long project, that simulates practical economic decision-making. Students started the semester with $10,000 and "ran their company" throughout the semester, applying the principles learned in class to a series of possible investments presented by the instructor. The game thus filled the role of the traditional homework problems, but had some additional goals and benefits:

➢ It created a framework for active learning of the material. Students had to not only learn various economic analysis techniques but also determine which ones were most applicable to the case at hand.

➢ It created a classroom environment that was fun, relaxed and informal while still being instructive.

The game was first developed and used at Rowan University in the spring 2001 and spring 2002 semesters. It was a highly popular and successful activity but required 6-8 hours per week of time on the instructor's part to run the game. Software has been developed, through an NSF-CCLI grant, which automates the game, requiring very little time on the instructor's part. The game was then integrated into a senior level course at the University of Kentucky in the fall 2003 semester. The software is currently being updated and debugged based upon the results of this trial.

This paper describes both the game itself and the software.

Description of Simulation

This section explains the mechanics of the game and provides details about some of the investment opportunities that made up the simulation. A more complete description has been published previously.[5,6]

Each student started the game with $10,000. The semester was divided into twenty turns. The stated goal of the game was to finish turn 20 with as much cash as possible. Thus, all investments had a fixed, known planning horizon (and no salvage value unless otherwise stated). Students were given the option of placing money in a savings account- with no minimum or maximum balance and no restrictions on frequency or size of withdrawals- at 5% interest per turn. In addition, students had the option of borrowing an unlimited amount of money at 15% interest, compounded every turn.

Many investment opportunities were introduced into the game through auctions. Students received specifications for each of the assets that would be auctioned off. They then applied present worth analysis to these assets and prepared a bidding strategy. Auctions were carried out using a sealed-bid format; each student submitted a bid without being able to see what others had bid. The asset was then awarded to the highest bidder at his/her bid price.

Other investment opportunities were introduced as fixed-price options (e.g.- do you wish to buy this for $1000, yes or no?) rather than through auctions. For example, in the "municipal bonds"

investment, students were given a list of several bonds with a variety of purchase prices, maturity values and maturity dates, and told they could purchase these bonds in any combination, but could not spend more than $2000. The purpose of the $2000 restriction was to create an example for which students would be rationing limited capital, regardless of their bank balance at that particular point. Other "fixed-price" investment opportunities introduced later included pirate ships, fishing boats, and ice cream trucks. Such examples were important to the game because they meant every student would definitely have the opportunity to make investments- relying on competitive auctions was not a necessity.

The complexity of the required analyses increased throughout the semester, reflecting new topics covered in class. For example, risk and uncertainty were introduced through several investment opportunities. The distinction between risk and uncertainty is that "risk" describes a situation in which multiple outcomes are possible but the probability of each is known. Risk was introduced into the game through examples such as this:

> A gold mine produces $500 of revenue per turn, starting the turn it is purchased, and continuing for an unknown period determined as follows. Each turn, the owner of the gold mine must roll two six-sided dice. If the total of the two dice is seven, the mine "craps out;" it yields $500 that turn but is worthless thereafter. On all other dice rolls, the mine remains productive the next turn and the dice are rolled again.

Examples like this were used as the first introduction to risk because the rolling of dice was a familiar everyday activity. Later, pirate ships were introduced into the game to illustrate more practical probability distributions. Students could purchase treasure maps that would allow them to "find a buried treasure," the value of which would be determined from a uniform probability distribution.

"Uncertainty," by contrast, describes a situation in which multiple outcomes are possible but the probability of each cannot be quantified. This was illustrated in the game through opportunities such as mines and factories. The owner of a mine could make money only by selling raw materials (iron, wood, clay, stone or crude oil) to the owners of factories. Factory owners were permitted to buy these raw materials for $5/unit in unlimited quantities from the instructor, however, mines were capable of producing up to 300 units/turn of these same materials for $100 + $2/unit. Thus, factory owners could buy these same raw materials from student-run mines at prices considerably below $5 and both parties would benefit. However, because the mines and factories were all auctioned off on the same day, it was impossible to negotiate *exact* terms prior to bidding on a mine or factory- one would not even know who to negotiate with. Further, there was more than one of each type of factory and mine in the game, and the total capacity of all mines was in some cases greater than the total demand for that raw material, and in some cases less than the total demand. These facts introduced elements of competition that made these investment opportunities more realistic and more challenging to analyze.

Web-Based Software
The software is written in ASP, primarily because it is a built in feature of Microsoft Windows and is a common language for developing programs of this nature, and is suitable for interacting

with data. Microsoft Access was chosen to create the databases that record the assets and actions of all students. Using this environment, a student can view all information regarding his or her company and enter instructions through the use of HTML style menus, and the database is updated automatically through the ASP scripts.

Table 1 summarizes the pages. All of these pages were constructed using a standardized HTML template. Most of the pages listed in the tables are single-function pages that the student can use to either view specific information (e.g., see all the raw materials currently stored in his/her warehouse) or complete a specific task (e.g., purchase a fishing boat.) The "Main" page, pictured in Figure 1, is essentially a menu that allows the student to navigate to these single-function pages, and access to it is password protected. The "Manager" page, which is also password protected, is only accessible to the instructor. This allows the instructor to set up the simulation at the beginning of the semester (e.g.- the number of items available in the economy is dependent upon the number of students in the class) and also has an "end turn" button. The instructor can specify whatever schedule he/she wishes (the Turn Sequence section below explains the schedule that was used in the fall 2003 offering) and simply needs to press this button at the appropriate times. This will run the script, described further in the following section, which processes all actions for the turn. Processing a turn required 6-8 hours of professor time when the simulation was run "by hand" in the spring 2001 and spring 2002 semesters, but requires only a few minutes with the software.

Turn Sequence
This section describes how the simulation progressed in the fall 2003 offering. The simulation consisted of 20 turns. In general, one turn was processed each week. Students could access the web site at any time during the week to review the status of their own companies, and perform actions like these:
- make bids in auctions
- set production rates in mines or factories
- propose transactions with other students
- respond to proposals from other students
- borrow money
- purchase raw materials

Some actions, like purchasing raw materials, take effect instantly- the student can immediately see their bank balance decrease and the new material added to their warehouse. Other actions only take effect at the end of the turn.

Every week, at a pre-specified time, the turn was processed. This simply requires running the "end turn" script, which performs actions like these:
- processes the auctions, awarding the assets to the winners and deducting the money spent from their bank balances
- virtually rolls the dice to see if the gold mines continue operating
- checks to see if anyone sent out a pirate ship or fishing boat, and determines the outcome if so
- determines the number of orders that all factory owners have to fill in the following turn

- calculates interest on all bank accounts

Thus, this script produces an updated summary of each student's assets as of the end of the turn. The students can then access this information through the web site and begin work on the next turn.

After turn 11, no NEW investment opportunities were introduced. Turns were processed three times a week for the last nine turns. This accelerated schedule was considered reasonable because the students by this time were familiar with the software, and because at this stage they were simply maintaining the assets they had acquired, with no new investments to analyze.

Integration of Simulation into Course

The simulation was integrated into a senior course on engineering economics at the University of Kentucky. Ten chemical engineering students participated. Dr. David Silverstein was the course instructor, but because this was a preliminary test of the software, the software was housed on a Rowan University server and administered by Dr. Dahm.

Participation in the simulation was a semester-long project worth 25% of the student's grade. However, the students were not graded strictly according to their final bank balance. Grades were based upon weekly submissions to the instructor in which they explained their decisions in running the company that week and the reasoning and calculations behind each. Thus, each student had a weekly opportunity to demonstrate his/her understanding of the course material. The basis of the grade was not success or failure, but the soundness of the approach. For example, if two students bid $3250 and $3255 respectively on a particular item, with essentially identical reasoning and calculations, then only the higher bidder would actually receive the item in the game, but both would be viewed equally for grade purposes.

Note, however, that the software does not require this or any other grading scheme. It facilitates running the simulation by collecting and processing students' instructions, but does not evaluate the students- any individual instructor using the software can choose whether, and how, to grade the activity.

Assessment of Simulation

Table 2 gives some results from a survey of the students who used the software in the fall of 2003. Overall, the students viewed the activity as exciting and effective for helping them learn the course material. The test in the fall of 2003 did reveal some minor bugs in the software:

- Some bids were not recorded properly in one auction, and consequently the wrong person "won."
- Some of the running tallies, like the expenses and revenues for specific turns, were reported incorrectly on some pages (though the bottom line, the new bank balance, was always correct.)

- A loophole existed that allowed players to avoid paying the $10 per turn fee for maintaining a building

During the fall 2003 trial, Dr. Dahm was able to modify the databases by hand to correct for such errors. The software is currently being revised to fix these bugs and make the software more user-friendly. The revised software will be tested in the fall of 2004, in an engineering economics course at Rowan University. After that it is expected to be ready for widespread dissemination.

Summary

The economic simulation described here has been successfully integrated into a course on engineering economics, using web-based software developed at Rowan University. It was popular with the students as a vehicle for presenting the material in an interesting way and emphasizing the practical value of the material. The software is currently being revised and debugged in response to feedback from students who used it in the Fall 2003 semester. The completion of the software will make the simulation available for dissemination to other universities, and allow its adoption with a minimum investment of time on the part of the instructor.

Acknowledgement

References

1. Newnan, D. G., Lavelle, J. P. *Engineering Economic Analysis*. Engineering Press, Austin, TX, 1998.
2. White, J. A., Case, K. E., Pratt, David B., and Agee, Marvin H., *Principles of Engineering Economic Analysis*, John Wiley & Sons, New York, NY 1998.
3. Sullivan, W. G., Bontadelli, J. A., Wicks, E. M., *Engineering Economy*, Prentice Hall, Upper Saddle River, NJ, 1993.
4. Young, D. *Modern Engineering Economy*, John Wiley & Sons, New York, NY 1993.
5. Dahm, K. D. "Interactive Simulation for Teaching Engineering Economics," *Proceedings of the 2002 ASEE Annual Conference and Exposition,* June 2002, Montreal.
6. Dahm, K. D., "Interactive Simulation for Teaching Engineering Economics," accepted for publication in *Journal of SMET Education,* Dec. 2003.

Biographical Information

Kevin Dahm in an Assistant Professor of Chemical Engineering at Rowan University. He received his B.S. from Worcester Polytechnic Institute in 1992 and his Ph.D. from Massachusetts Institute of Technology in 1998. His current primary teaching interest is integrating process simulation throughout the chemical engineering curriculum, and he received the 2003 Joseph J. Martin Award for work in this area.

Ravi P. Ramachandran is an Associate Professor in the Department of Electrical and Computer Engineering at Rowan University. He received his Ph.D. from McGill University in 1990 and has worked at AT&T Bell Laboratories and Rutgers University prior to joining Rowan.

David L. Silverstein is currently an Assistant Professor of Chemical and Materials Engineering at the University of Kentucky, assigned to the College of Engineering Extended Campus Programs in Paducah. He received his B.S. Ch.E. from the University of Alabama in Tuscaloosa, Alabama; his M.S. and Ph.D in Chemical Engineering from Vanderbilt University in Nashville, Tennessee; and has been a registered P.E. since 2002.

Table 1: List of all ASP pages and their functions.

AcceptOrder	Allows you to accept or decline an offer from another student.
Auction	Indicates what is available in current and future auctions, allows students to enter bids.
Borrow	Asks the user to confirm a request for the user to borrow funds from the Bank of Dahm.
BuyBoats	Asks the user to confirm an order for a boat. Also checks to see if the user has the necessary funds
Checkpword	Checks to see if the password the user entered is correct. If it is, the user can enter the game, if not the user is told that the password is incorrect
CompleteBoat	Notifies the user that they have bought a boat, removes the money from their account and places the order for the boat.
EndTurn	Processes results when Manager has indicated the turn is over.
Fish	Allows a student to organize fishing and treasure-hunting expeditions.
Gofish	Processes fishing expeditions, determines revenue for each boat.
Main	Menu providing access to all other functions.
Manager	Asks the instructor to confirm password.
Manager2	Menu page for instructor actions.
Ordercomplete	Processes the orders for raw materials and sends a message to the user confirming their order
Orders	Allows a student to view transactions that have been proposed to you, accept or decline, and make proposals of your own.
Password	Asks the user to confirm their password before they are allowed into the game.
Placebid	Allows student to enter a bid for an auction and confirms the bid has been entered.
PlaceOrder	Processes a proposal made by one student to another- the other student accepts or declines using the AcceptOrder page.
Produce	Allows a student to specify rate at which materials are produced this turn in mines and factories.
Purchase	Allows student to request purchases of raw materials at the flat rate of $5/unit.
Removebid	Processes a request to cancel an auction bid and confirms this has been done.
Result	Checks to see that the login name/password are valid, returns "yes" or "no."
SellGoods	Allows a student to specify the price at which products are to be sold next turn.
SetProduction	Splash page that confirms the results from the "Produce" page.
Ships	Processes the purchase of pirate ships and fishing boats from the shipyards.
SimpleForm	The student login page.
Sold	Splash page that confirms that a new sale price has been set for a product.
Stats	Shows a summary of the student's current assets and liabilities.
Transaction	Handles borrowing money from the bank.

Table 2: Summary of Student Feedback (5=strongly agree, 1=strongly disagree)

Question	Mean Response
The game helped motivate me to learn the course material.	3.7
In order to do well in the game, you needed to have a good understanding of the course material.	3.9
I would recommend that future offerings of the course continue to use the game.	4.1

Figure 1: Screen capture of "Main" page.