CHANGES IN STUDENTS' PERCEPTIONS OF PROGRAMMING SKILLS USING CLASS PROJECTS

Lori Baker-Eveleth, Robert W. Stone, and Norman Pendegraft, Department of Business, P.O. Box 443161, College of Business & Economics, University of Idaho, Moscow, ID 83844-3161, *leveleth@uidaho.edu, rstone@uidaho.edu, norman@uidaho.edu*

ABSTRACT

Information system (IS) students typically complete a computer programming course as part of their curriculum. To further enhance their skills, incorporation of programming projects in the upper division IS courses were assessed. The impacts of the projects on students' perceptions of their programming skills used the theoretical foundation of self-efficacy theory. Results from the estimation showed that students' positively changed their perceptions of their programming ability when impacted by changes in intellectual interest and encouragement of peers and mediated by changes in self-efficacy and outcome expectancy.

THE THEORETICAL MODEL

As information systems (IS) faculty our goal, for undergraduate students, is to develop IS students' skills for a successful transition into the work environment. A strong foundation in logic and programming allows migration and adaptation of IS skills to a variety of business contexts. Encouraging students to continue developing their logic and programming skills necessitates faculty consider changing the IS curriculum. Students were required to complete projects in six IS courses with an objective of developing their programming and logic skills. The objective was to reinforce and improve students' knowledge of programming and logic as the student progresses through the curriculum rather than compartmentalizing these efforts in the required programming class.

The model used to guide the development of a foundation of programming and logic skills in the IS curriculum is based in self-efficacy theory. The theory [3, 2] links an individual's cognitive state to a variety of affective and behavioral outcomes and perceptions of future outcomes (i.e., loss of control, low self-confidence, low achievement motivation) [12]. Self-efficacy theory has also been used to explain user reactions to information technologies [3, 4, 6, 7, 10, 11], inclusive of programming and logic skills.

According to self-efficacy theory expectations (e.g., motivation, performance, and feelings of frustration associated with repeated failure), in large part, determine affect and behavioral reactions in numerous situations. Four groups of variables or experiences identified by Bandura [1] impact an individual's self-efficacy evaluations of a specific task. The strongest is the individual's personal mastery or accomplishments regarding the task. Prior success at performing a task increases self-efficacy of that task. On the other hand, failing repeatedly at performing a task lowers these expectations [5]. Modeling the behavior of others who successfully completed the task or vicarious experience is the second group of variables. The observer can improve their own performance through observing others successfully completing the task [1, 5]. Social persuasion is the third group of antecedents to self-efficacy and outcome expectancy. Social persuasion occurs when individuals are led or have it suggested to them that they can successfully complete the task in question. Common forms of social persuasion are verbal encouragement, coaching, and providing performance feedback [1]. The last grouping of antecedent variables is physiological arousal and emotional states. Physiological arousal of the individual impacts their expectancy judgments regarding specific tasks [1]. Improving perceptions of self-efficacy occurs when there is intellectual interest in a task. Negative judgments of one's efficacy can be produced from anxiety regarding a specific task [3].

Based on the literature presented above, a model was developed. This model relates the antecedents of self-efficacy and outcome expectancy to students' perceptions of their programming skill, mediated by self-efficacy and outcome expectancy. Because the focus of the research is the changes in students' perceptions of their programming skills due to changes in the model's variables as a result of completing IS projects in upper division courses, the model is presented in this differential format.

THE EMPIRICAL STUDY

Data collection was conducted by a pre- and post-survey of students. The six courses with a programming project identified were the target population. The total response rate for the survey was 78 participants with 41 responses completing either the pre- or post-survey but not both. Therefore, the number of participants who completed both the pre and post project questionnaire was 37. These observations were used in the empirical analysis presented below. All the statistical analyses were performed in PC SAS version 8.2. The target population's characteristics are shown in Table 1.

Table 1. Study Characteristics

Characteristic	Average	Major	Percent
GPA	3.13	Computer Science	2.70%
AGE	23.60	Information Systems	37.84%
Female	35.14%	Information Systems & other major	56.77%
Male	6.86%	Other/undecided	2.70%

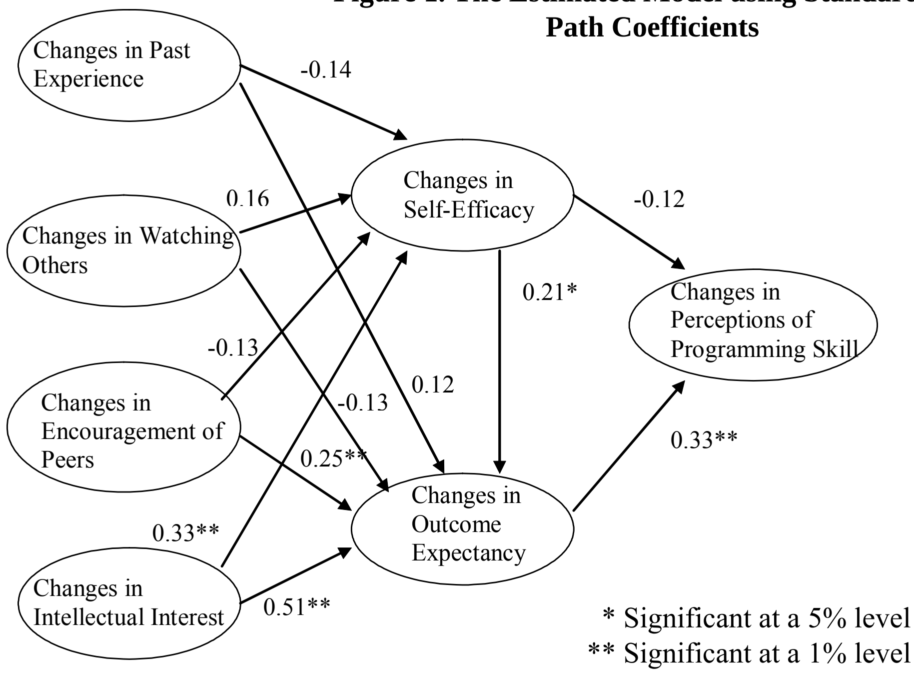
The measures of the constructs in the model were developed in the context of a generic IS project. These measures were formed by summing the respondents' answers to selected questionnaire items. The items were summed due to the relatively small sample size of the data set and the estimation technique employed. The individual items are shown in Table 2 grouped in their construct measures. The measures of self-efficacy and outcome expectancy were modified from previously published work by Henry and Stone. Self-efficacy was developed based on Henry and Stone [9], while outcome expectancy was developed from Henry and Stone [8, 9]. The remaining measures were developed by the authors. For all items, respondents were given a five-point Likert-type scale upon which to respond. The response options provided were strongly agree (5), agree (4), neutral (3), disagree (2), and strongly disagree (1).

The Estimation of the Model

The estimation of the theoretical model was performed using the mean differences of the summated measures for the 37 paired observations. Each measure represented the construct with the same name in the theoretical model. The estimation approach was system of simultaneous equations using CALIS (i.e., Covariance Analysis of Linear Structural Equations) in PC SAS version 8. The estimation method used was maximum likelihood. The goodness of fit index was 0.95 and adjusted for degrees of freedom it was 0.66. The root mean square residual was 0.06.

The first paths discussed are those with significant, positive impacts on students' perceptions of programming skills. The estimated model is shown in Figure 1. Changes in intellectual interest from

Figure 1. The Estimated Model using Standardized Path Coefficients



completing the IS project lead to positive changes in self-efficacy. Other changes were noted in encouragement from peers and intellectual interest leading to positive changes in outcome expectancy. The self-efficacy changes led to changes in outcome expectancy. It is also the case that changes in outcome expectancy produced changes in students' perceptions of their programming skills.

The antecedents of changes in self-efficacy and outcome expectancy were also allowed to pair-wise correlate. Two pairs of these variables had significant correlations: changes in encouragement from peers (i.e., social persuasion) and changes in watching others (i.e., vicarious experiences), and changes in past experience (i.e., personal mastery) and changes in watching others. All these correlations were positive in direction.

DISCUSSION OF THE RESULTS

The programming projects implemented across the curriculum enabled changes in students' perceptions of their abilities. The result of changes in intellectual interest positively changed students' perceptions of programming skills through changes in self-efficacy and outcome expectancy. This confirms what the authors expected. If the IS project is appropriately challenging, after completing the project these students have heightened intellectual interest which positively changes their self-efficacy, outcome expectancy, and ultimately perceptions of their skills. This result reinforces the adding of projects with programming components to a series of courses in the IS curriculum. The desired outcome was to improve students' actual programming skills and their appropriate perceptions of their skills. The empirical results indicate that by completing these projects, students' perceptions of their skills did improve. Thus, instructors can impact students' perceptions of programming skills through requiring appropriate class projects.

A related result is the influence that changes in intellectual interest have on changes in outcome expectancy. These factors ultimately change the perceptions of students' programming skills. As with the earlier discussed result, the implication of this result lies in the subject and design of the class programming project.