Improving Computer Science Education Through Cloud Computing: An Observational Study

Jorge Edison Lascano[1,2]
[1]Departamento de Ciencias de la Computacion
Universidad de las Fuerzas Armadas ESPE
Sangolqui, Ecuador
edison_lascano@yahoo.com

Stephen Clyde[2]
[2]Computer Science Department
Utah State University
Logan, USA
Stephen.Clyde@usu.edu

***Abstract*—In an earlier study, we observed that students in a small graduate class who used Cloud Computing (CC) for their programming assignments improved their analysis-to-reuse (A2R) skills more than students who didn't use CC. That preliminary result motivated us to see if the use of CC in programming assignments would yield similar results for a broader range of classes and students. To this end, this paper reports on an observational study on the students of the Computer Science Department of Utah State University that spanned from August 2015 to December 2016 and included over 221 students, with data collected at three different times. An ANOVA statistical analysis of the study data revealed a significant difference in the perceptions about acquired A2R skills in favor of students that used CC**

Keywords—computer science education, software engineering education, programming, cloud computing, observational study

I. Introduction

Cloud Computing (CC) is changing the way IT users consume computing resources. CC introduces new elements for execution and development environments, these resources are consumed and shared on demand among software systems stakeholders. Currently, higher education institutions are including CC in their curricula to improve their students' skills [1], and careers have been created around this technology [2] [3] [4] [5] [6] that many organizations have adopted or are moving towards its adoption [7]. CC is currently used in different areas such as infrastructure provisioning, test and development, file storage, disaster recovery, and backups [8]. Most of the industry already requires professionals to have a foundation in the CC body of knowledge, adopted in 2008 by IEEE [9]. This trend represents both an opportunity and a challenge for higher education. Specifically, higher education needs to (a) incorporate CC technology into curricula so students can be better prepared, (b) understand its impact in higher education [10], and (c) leverage CC technology as means of helping students improve other skills and knowledge required for Computer Science (CS) professionals.

Although there exist a wide range of literature in this aspect [11] - [22] that cover the use of CC for different courses and the inclusion of CC as a solution for higher-education infrastructure, none of them use CC throughout the full software development process nor do they don't analyze the impact of CC on improving students' A2R skills using CC resources for programing assignments.

CS departments need to improve their students' A2R skills to keep pace with industry demands. Also, they want to comply with accreditation criteria. In the United States of America, the Computing Accreditation Commission of ABET (Accreditation Board for Engineering and Technology, Inc.) [23] is responsible of accrediting the CS programs. ABET establishes general criteria that apply to all programs, and program criteria that apply to a specific program. These general criteria cover program educational objectives (PEO), student outcomes, continuous improvement, curriculum, faculty, facilities, and institutional support. The CS Department of Utah State university has defined the following as one of its PEOs:

PEO-1: "The USU Computer Science program will prepare its graduates to be successful and contributing professionals by being able to apply the principles of computer science and adapt emerging technologies to analyze and solve real world problems" [24].

A preliminary study conducted in 2015 [1], where the instructor of CS6200 from Utah State University used Amazon Web Services for advanced distributed systems programming assignments, showed that CC helps students improve A2R skills. The results of this preliminary study motivated us to plan a new study, encompassing as much of the CS Department as possible, to see if the conclusions would hold for a broader Group and different classes. Section 2 provides necessary background information for this observational study and Section 3 summarize a brief literature review on the use of CC in educational institutions.

This observational study aims to find statistically significant differences in the perceptions of students with respect to their A2R skills between those who used CC in their assignments (Group A) and who did not (Group B). To this end, it focused on two questions: (1) what are students' perceptions about how their own skills and knowledge change over the course of a semester and (2) how does the integration of CC into programming assignments affect those perceptions? Section 4 describes the study's design and the survey instrument used in the study to gather data about student perceptions relative to these two questions. Section 5 provides details about the results

2377-570X/17 $31.00 © 2017 IEEE
DOI 10.1109/CSEET.2017.33

of the quantitative analysis. Finally, section 6 presents our conclusions and future research.

II. Background

A. Cloud Ccomputing Services

CC, which "provides shared computer processing resources and data" [25], is a technology trend that is taking off, especially since CC providers now offer services that help organizations overcome their security and compliance concerns [26]. CC is widely becoming an integral piece in the software development process and an important component in complete software solutions. Even IEEE has considered the importance of CC in professional and academic environments and founded in 2011 a global initiative, IEEE Cloud Computing, to promote CC and its related technologies [27]. This initiative comprises standards, publications, education, careers and conferences.

CC providers offer their services under different models [25], three are the standard models as defined by the National Institute of Standards and Technology (NIST): Infrastructure as a Services (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). IaaS is used for deployment and execution of computer systems, IaaS can greatly simplify an organization's IT management and support on-demand scalability, high availability, fault tolerance, and disaster recovery [28]. PaaS provides developers with full out-of-the-box development environments, PaaS solutions typically include virtual machines, operating systems, programming tools, and databases [29]. A third category of CC services is SaaS, solutions in this category are target as end users and cover a wide range of application software. The SaaS users don't need to purchase or install application software locally. Instead, they use existing applications in the cloud. CC services provided by Amazon Web Services (AWS) [30] include: virtual instances – EC2 (IaaS), code repositories – CodeCommit (SaaS), code test and deployment – CodePipeline (SaaS), Databases –RDS and DMS (PaaS), scalable storage –S3 (IaaS), big data processing – EMR (PaaS), and web scalability – AWS Elastic Beanstalk (IaaS). Providers such as Google, IBM, Microsoft, and others offer similar CC solutions.

Since cloud services are widely used by enterprises, cloud providers offer their services to educational institutions at no cost or at very low cost, so the new professionals can learn, and test their services. Examples of these programs include Amazon Education [31], Google for Education [32], GitHub Education [33], IBM Cloud Education [34], and Azure in Education [35]. Most of them provide biannual grants to students or instructors.

B. Observational Studies

Experiments are used to search for cause and effect relationships [36]. Researchers design experiments to predict what can happen by varying some values and observing changes. Variables are used to affect and to quantify these changes and may be any factor, trait or condition. Two important types of variables are independent and dependent variables. The former are changeable or controlled by the scientists. The latter represent the outcome in function of the independent variables [37]. An observational study [38] draws inferences from a sample to a population where the researcher does not have full control over the independent variables [39]. Some reasons for using an observational study instead of a controlled experiment include: the need to respect human rights and logistical issues. Also, sometimes it is simply impossible to control the independent variables sufficiently.

As mentioned before, certain independent variables will be outside the control of the researcher, but they can be observed and recorded. Cause and effect are difficult to establish in observational study, nevertheless, they can allow researchers to formulate some associations and lay the foundation for future studies that can be carried on in control settings [40]. According to Shull et al. [41], after initial feasibility studies, researchers can use observational studies to collect data that will help explain the considered phenomenon and "formulate hypotheses to be tested in subsequent experiments" [42]. "A well designed observational study, resembles, as closely as possible, a simple randomized experiment" [38]. The main difference is the randomization of an experiment, where participants are selected by chance, so bias can be reduced.

III. Related Work

The use of virtual resources for CS courses is not a new concept in the academia [11]. Studies report that the use of CC in the class has proved to be worthwhile, by allowing students to improve their professional skills and to obtain a better understanding of realistic execution environments and issues in areas such as security [12] [11], networking/network programming [13] [14], system and network administration [11] [15], distributed systems [16], and data processing [17] [43]. Other researchers have investigated the relationship between CC and higher education, including CC adoption and its influence [18] [19], students perception of CC effective use [20], CC impact [21], and relevance of CC [22].

Gonzalez et al. for example, at Rochester Institute of Technology used Amazon EC2 in their Principles of Systems Administration course to leverage career opportunities for their students [11]. Zhu used cloud resources in the Network Programming course at Metropolitan State University of Denver. The students implemented four cloud-based network programming projects and every student was able to use multiple virtual machines (EC2 instances) provided by AWS. Based on CS3700 students' perceptions. In general, the students agreed that using CC resources had a positive impact on their learning experience [44] and Zhu states that AWS was helpful for their learning and career development. Zhu also claims that the same effect should be true for other courses, but unfortunately there is no other studies published to date that can substantiate his claim. Rabkin et al. [43] used CC for MapReduce measurements at University of California, Berkeley, concluding that there is a need for students to experience running and debugging distributed applications in a realistic infrastructure.

Besides using public clouds, some research universities such as Syracuse [13], North Carolina State [16], Stony Brook [45], Arizona State [14], among others, have implemented their own clouds using campus resources and have made them available to their students and to their departments. These internal clouds have the benefits of 1) using virtually local resources as clouds, 2) giving students opportunities to learn about the implementation of clouds, and 3) expanding education infrastructure within their institutions. Nonetheless, implementing private clouds for most institutions is not economically convenient [44], especially when cloud-service providers offer academic grants to students and faculty. Consequently, universities tend to favor public cloud services for educational purposes.

IV. Design of The Observational Study

In [1], we described our findings on the use of Cloud resources for programming assignments. Students that used CC for their assignments improved their A2R skills in a level higher than those ones that did not use CC. This group of students corresponds to the CS6200 course taught during Spring-2015, where participants developed advanced distributed applications, some of them used IaaS and PaaS, others did not use it. After this experience, we aimed to extend our work to other programming courses, by including CC as a resource for students' assignments during Spring-2016 and Fall-2016 semesters. With the hoped-for a bigger improvement on A2R skills for these students than for the Fall-2015 students, who did not use CC resources.

We defined three main variables for our experiment, the use of CC (the treatment) as the independent variable, and the variations in A2R skills and knowledge levels (covariates) as the dependent variables. Nonetheless, it was not possible to prepare a properly randomized selection of students who will or will not use CC in assignments at the CS department. So, the independent variable is outside our direct control. Alternatively, we requested the instructors of CS3450, CS5110, CS5200, CS5600, CS5680, CS5700, CS5800, CS5890, CS6110, CS6600, and CS7910 courses to use CC in their assignments. During the Spring-2016 semester, the instructor of CS5200 agreed to use CC, and during the Fall-2016 semester, the instructors for CS3450, CS5600, CS5700, and CS6600 agreed to the request. Students in all other classes would continue not to use CC in their programming assignments.

The introduction of CC in programming assignments was a straightforward process: 1) The instructor designs the assignment without CC resources in mind. 2) The instructor determines what kinds of resources could be used, for example, to develop a program, they could use a PaaS resource instead of their local development software. To deploy their programs, they could use IaaS resources such as virtual machines in the cloud instead of their laptops or lab computers. 3) The instructor applies for Grants accordingly, for example AWS, at the time of this research, offered $100 dollar-Grants for students to consume their cloud services, there is also a set of free-tier services from different vendors and students can apply for educational accounts. 4) Finally, the instructor sets up the services and integrate them into the assignment descriptions and materials.

The most of the CC services that ended being used were IaaS, such as AWS EC2 virtual servers and Amazon S3 storage. Programming classes did used some simple SaaS services, like Bitbucket and GitHub, for Git repositories. Requiring instructors to use CC was not possible and following up with the students who used CC was challenging. So, we let the instructors use CC on their discretion and made sure the survey instrument would collect sufficient information to determine whether they used CC in their programming assignments, as well as their perceptions about their levels of skill and knowledge.

Furthermore, we designed the surveyed instrument overall curriculum objectives in mind, like PEO-1 and the following student outcomes from ABET's guidelines [46]:

(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

(c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

(i) An ability to use current techniques, skills, and tools necessary for computing practice.

(k) An ability to apply design and development principles in the construction of software systems of varying complexity.

To organize the survey questions, we first decomposed PEO-1 into three sub-objectives (Fig. 1): Analyze and solve real world problems, adapt emerging technologies, and apply principles of CS. The first two sub-objectives deal with A2R skills level, while the third deals with general knowledge. Next, we aligned the sub-objectives with the student outcomes and then decomposed the outcomes into eight areas: Problem Analysis, Requirements Identification and Definition, Systems Analysis, Systems Design, Current Practices, Other Software Engineering Skills, Tools, and Principles. The questions were designed with these eight areas in mind, following the Goal Question Metric paradigm [47].

The first four questions of the survey though, are not meant to measure any level of knowledge or skills, they are intended to capture the characteristics of the participant and their classes, i.e., student number, age, past classes, and current classes.

Question 5 captures the students' perception of A2R skills level, while Q6 measures the students' perception of knowledge. Each of the questions contain several sub-questions or topics and asked the participants to rank their perceptions of their own skills and knowledge on a scale from 1 to 5, relative to both the beginning and at the end of every period, for each one. Each topic has a *N/A* option for students who don't recognize the topic or feel that it doesn't apply to them. See Fig. 2 for a snippet of Question 5 that shows just two topics.

Question 7 is about design principles and assignments characteristics. Questions 8 and 9 of the survey were open so the students can express themselves about how they think that their skills can be improved and about what methods or tools would help them to adapt to emerging technologies.

The study design originally called for a survey to be conducted using this instrument at the end of every semester throughout the study period. However, because of semester breaks, it became necessary to administer the surveys at the beginning of the following semester.

While planning the study that include a group of human beings (the students), we applied for and obtained the necessary Institutional Review Board (IRB) approvals, and then we started

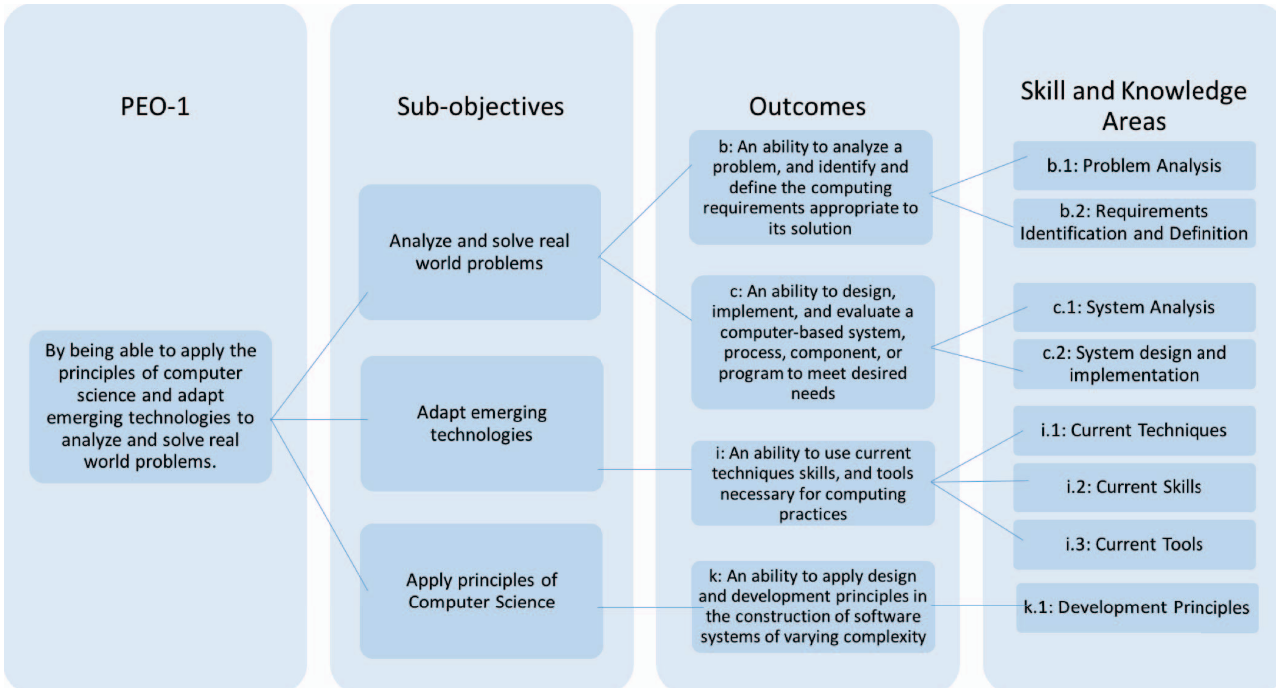

Fig. 1. Goal Question Metric applied to PEO-1

collecting data from the students of the CS department of Utah State University at the end of Fall-2015 semester. Then, we needed to measure the difference in perception of the students, and we collected data from the next two contiguous semesters: Spring-2016, and Fall-2016. The study intended to aim about 650 students of the CS department, i.e. students that took CS classes in any of the Fall-2015, Spring-2016 and Fall-2016 semesters, a total of 221 different students responded the survey, some of them, as expected, retook the survey in following semesters. Our goal was that at least 20% of the 650 students respond every semester, i.e., 130 students. For the Fall-2015 semester, there were 153 respondents to the survey, for Spring-2015, 181 and for Fall-2016, 117.

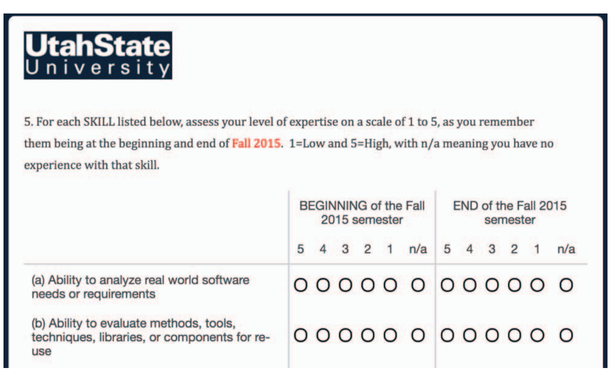

Fig. 2. Snippet of Question 5 from the Fall-2015 semester survey

Before we started processing the data, we needed to clean it. Incomplete surveys were dropped. Meaningless survey data such as answers with the same level of perception for all questions were eliminated. Surveys with duplicated students' id numbers for the same semester were also removed.

Significant tests computed the probability for our hypothesis, H_1: *Students who had assignments that required the use of CC have a greater perceived increase in her/his skills and knowledge levels in the following A2R areas: tool evaluation, development environment setup, runtime environment configuration, analysis, design, application of "best practices", testing, deployment, and reuse.*

These questions/metrics (Table 1, Table 2) capture students' A2R skills level in a scale from 1 to 5 at the beginning and end of a semester, the difference $\Delta A2R = A2R_{After} - A2R_{Before}$ represents students' capacity variation on their skills in the course of a semester. Gelman [48] suggests that survey questions' weights depend on the actual data and the survey design. We need to have a total measurement of the A2R skills composed by the 23 sections of question 5. Then, we gave arbitrary weights to every section of the survey based on their importance to the outcome they are dealing with, i.e. to the ABET PEO-1 decomposition, and to industry requirements. We considered 1.0 as the highest weight according to its subjective importance in relation to students' perceptions.

The main purpose of this observational study is to analyze the perceived impact of using CC in programming assignments for improving PEO-1. For this reason, we compared statistically $\Delta A2R$'s of students that use CC (Group A), and students that do not use CC (Group B), $\Delta A2R_A$ vs $\Delta A2R_B$. Finally, we define our null hypothesis, H_0, that states: *The use of Cloud Computing resources for programming assignments does not have any effect on the incremental improvement of A2R skills for students.*

TABLE 1. WEIGHTS FOR QUESTION 5: PERCEPTIONS OF LEVEL OF STUDENTS' A2R SKILLS

#	*A2R Question*	*Weight*
a	Ability to analyze real world software needs or requirements	0.6
b	Ability to evaluate methods, tools, techniques, libraries, or components for re-use	0.1
c	Ability to reuse methods, designs, or software from previous assignments	0.8
d	Ability to design a software system to meet complex real-world requirements	0.8
e	Ability to implement a software system according to a design	0.8
f	Ability to implement software that can run in a runtime environment different from your own computer or one in a school lab	0.4
g	How to configure a runtime environment into which you can deploy a software system that you built	0.6
h	Ability to deploy a system to a runtime environment different from your own computer or one in a school lab	0.3
i	Ability to learn and use virtual or cloud-based resources for creating software solutions	0.6
j	Ability to thoroughly test software systems using executable test cases	0.8
k	Ability to think critically and develop alternative solutions to a problem	0.6
l	Ability to think creatively about software solutions	0.8
m	Ability to understand, evaluate and use emerging technologies	0.4
n	Ability to learn and use existing software services available on the Internet	0.4
o	Ability to follow industry-wide "best practices" when use your chosen development environment	0.5
p	Ability to follow "best practices" in testing	0.8
q	Ability to follow "best practices" in deploying software to runtime environments	0.4
r	Ability to apply green practices	0.1
s	Ability to evaluate a variety of operating systems and frameworks as possible runtime environments	0.1
t	Ability to apply collaborative methods, tools, techniques to develop software	0.6
u	Ability to design a software system that will provide for a good user experience	0.6
v	The ability to create maintainable software	0.6
W	Ability to understand real-world problems related to the course material	0.8

V. DATA ANALYSIS AND INTERPRETATION

The data collected throughout the three semesters span was compared from different perspectives and for different groups of students. For every comparison, an analysis of Variance was performed using ANOVA for a single factor, t-test or z-test. We gathered a total of 451 responses from the three surveys, 280 were answered in their entirety, and 264 were considered valid, 95 for Fall-2015, 102 for Spring-2016, and 67 for Fall-2016. From the 264 students, only 37 would have taken courses that include the use of CC in their programming assignments, leaving a total of 227 students that did not have exposure to CC. There are 16 surveyed students that took either CS5200 or CS5700 when CC was not used in their assignments during Fall-2015; and there are 12 students that took CS5200 or CS5700 when CC was introduced as a resource for programming assignments during Spring-2016 and Fall-2016, Table 3 summarizes the values statistically obtained, namely: Averages, p-values, and Differences.

We run a qualitative analysis by gathering data about the level of perception of every student and assigning an arbitrary weight to every question. We relied on the analysis of variance to proof the non-validity of our Null Hypothesis, H_0.

For this case study, we concluded that students that use CC resources for their programming assignments improved the perception of A2R skills in a higher level than students that did not use it, obtaining an improvement on PEO-1. This inference is supported by calculating the probability for H_0. We analyzed the differences between Group A and Group B ($\Delta A2R_A$ *vs* $\Delta A2R_B$). We got a p-value = 0.014; as p-value < 0.05, H_0 is rejected in favor of H_1.

TABLE 2. WEIGHTS FOR QUESTION 6: PERCEPTIONS OF LEVEL OF STUDENTS' KNOWLEDGE

#	*Knowledge Question*	*Weight*
a	Understanding of how to match the needs of an application to an appropriate develop environment and runtime platform	0.8
b	Understanding of at least one development stack (a collection of reusable components or libraries)	0.7
c	Understanding of development environment setup	0.6
d	Understanding of principles of software testing	0.8
e	Understanding of network communications	0.4
f	Understanding of the principle of reliability as it applies to software systems	0.7
g	Understanding of security principles and practices for software systems	0.7
h	Understanding of what affect runtime performance and how to detect inefficiencies and correct them	0.5
i	Understanding of the principle of Coupling	0.5
j	Understanding of the principle of Cohesion	0.5
k	Understanding of software reuse	0.8
l	Understanding "abstraction" with respect to the design and implementation of software system.	0.6
m	Understanding "encapsulation" with respect to the design and implementation of software systems	0.6
n	Understanding "modularization" with respect to the design and implementation of a software system	0.6

Students of CS5200 and CS5700 courses were special groups that in previous semesters (S_{i-1}) did not use CC in their assignments and in following semesters (S_i) use CC resources. Their perceptions show similar results than the comparison between Groups A and B. with a well noted tendency to a higher increase, 0.426, for Group $A_{5200/5700}$, versus 0.262 for Group $B_{5200/5700}$. In this analysis H_0 was strongly rejected in favor of H_1 with a p-value=0.00959 < 0.05, that is a positive influence of using CC improve students' A2R skills. These results corroborate the rejection of H_0, nevertheless a different environment setup may be needed for future experiments where in the same course, half of students use CC and the other half in the same class do not. Another propitious environment would be two courses of the same subject in a concurrent semester, one course using CC and the other not using it.

TABLE 3. ANOVA P-VALUES FOR A2R SKILLS VARIATIONS (QUESTION 5) FOR STUDENTS USING CC (A) AND STUDENTS NOT USING CC (B)

Comparison (A=use of CC, B=no use of CC)	***Groups according to Question 5***	***Count***	***Average***	***P-value***	***Diff.***	***YES/NO statistically significant difference (p-value<0.05)***
ΔCC_A vs ΔCC_B	Q5i GA diff	37	1.1622	0.0215	0.4573	YES
	Q5i GB diff	227	0.7048			
$(A2R_A$ vs $A2R_B)_{End}$	Q5 GA End	37	3.3366	0.2477	0.1692	NO
	Q5 GB End	227	3.1674			
$(A2R_A$ vs $A2R_B)_{Beginning}$	Q5 GA Beg	37	2.3523	0.5424	-0.0967	NO
	Q5 GB Beg	227	2.4491			
$\Delta A2R_A$ vs $\Delta A2R_B$	*Q5 GA diff*	*37*	*0.9842*	*0.0141*	*0.2621*	*YES*
	Q5 GB diff	*227*	*0.7220*			
$\Delta A2R_{A-5200/5700}$ vs $\Delta A2R_{B-5200/5700}$	*Diff A2R CC*	*12*	*1.1041*	*0.0096*	*0.4265*	*YES*
	Diff A2R No CC	*16*	*0.6775*			
Same Students F15 – S16	Q5 F15 After	30	3.2056	0.0005	0.8213	YES
	Q5 S16 Before	30	2.3843			
Same Students S-16 – F16	Q5 S16 After	16	3.1882	0.0633	0.6928	NO
	Q5 F16 Before	16	2.4954			

VI. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we presented the results of a one-and-half year observational study about the use of CC in programming assignments for a group of CS students, concluding that the use of CC has a positive impact. Specifically, our findings confirmed our hypothesis that students exposed to CC (Group A) in their assignments have a bigger increase in their perception of acquired A2R skills than students who did not use any CC (Group B) in their assignments, as it can be seen in Table 3, there is statistically significant difference (p-value=0.0141) between Group A and Group B. Therefore, there is a likelihood that the use of CC in programming assignments for CS students has a positive influence in increasing students' A2R skills.

We tested for significance using the Analysis of Variance Algorithm (ANOVA [49]) with a probability (p-value) cutoff of 0.05. In most cases, we obtained a p-value < 0.05 meaning that there is statistically significant difference [50] [51] between the group that used CC and the group that did not use CC. Hence, we rejected H_0 in favor of the alternate hypothesis, H_1, i.e., students who were exposed to CC ($\Delta A2R_A$) noticed a bigger increase in their perceptions of acquired A2R skills than students that were not exposed to CC ($\Delta A2R_B$).

This study's findings, mainly the proved non-validity of the null hypothesis based on a p-value = 0.014, open doors for future experiments. An interesting hypothesis for follow-on experiments would be "the use of CC can help improve communication skills for team projects assignments." An experiment would consider students in a software-engineering course complete assignments in groups instead of individually. Some course sections would use CC and others would not. A more complex experiment would involve the continuation of projects in a course sequence, e.g., one course could model a system and subsequent courses would then implement that model. A couple of sections would use CC services, such as virtual servers and code repositories, other sections would use nonCC development tools and share code by copying files from computer to computer.

Also, future experiments could look at even broader populations and group randomization techniques. For example, group selection would be better if we could split students within a single class into A and B groups. Unfortunately, that may interfere in the learning process or create a lot of extra work for the instructors. For example, we may need to plan recovery courses for students that show a lower increase on their skills through the experiment.

The survey used in this study contained a couple open questions, namely Questions 8 and 9. A future research project could use sentiment analysis [52] to study the qualitative data that were collected for these open questions. This could provide some insights for how the students feel about their skills and their knowledge. An informal analysis of these data shows that the most used words are "software", "real" and "world". From here, the sentiment analysis would try to determine whether these ideas are connected to positive, negative and neutral attitudes.

Uncertainty of the advantages of CC in education still exists, even though enterprises of all kinds are currently using CC to build and deploying software. One disadvantage, is that for novice developers, like students, CC services are just one more complexity that they must learn and it could easily become a stumbling block. Any integration of CC in a CS curriculum must be well thought out and aim to eliminate or minimize these obstacles.

Since this research was an observational study and we could not control the group selection, we ended up with widely different group sizes: 37 in Group A and 227 in Group B. Ideally, the size of each group should be the same. Nevertheless, ANOVA allows for uneven group sizes. Other comparison sets were found among the students of contiguous semesters for the same subject. For example, CS5200 and CS5700's students that did not use CC during the semester S_i, i.e., 16 from Group B, versus the same courses' students that used CC during the semester S_{i+1}, i.e., 12 from Group A. However, this comparison introduction would add extra variables, such as variations in commitment in faculty members and motivation for students to be part of this study.

The adoption of CC for programming assignments in the research population of USU was not widely accepted, instead instructors were inclined to reject its use, a future study could analyze the causes and propose a better approach such as the Technology Acceptance Model (TAM) which focuses in the perceived usefulness and perceived ease-of-use of new technologies.

We performed a statistical analysis of Students' perceptions of A2R skills and knowledge levels at the end of Fall-2015 versus the beginning of Spring-2016. Same analysis was performed at the end of Fall-2016 versus the beginning of Fall-2016. Although we thought that their perception should not be statistically significantly different at the end of semester S_i than at the beginning of semester S_{i+1}, the results, see Table 3, showed that they are statistically different. Future studies may help determine better methods to help students retain their knowledge and skills between semesters' breaks.

Acknowledgment

We thank Seyed Mansourbeigi for assistance with the statistical analysis.

References

[1] J. E. Lascano and S. W. Clyde, "Using Cloud Services To Improve Software Engineering Education for Distributed Application Development," presented at the ICSEA 2015, The Tenth International Conference on Software Engineering Advances, 2015, pp. 438–444.

[2] "Cloud Computing and Virtualization | Masters of Science in Software Engineering | SJSU." [Online]. Available: https://msse.sjsu.edu/cloud-computing-and-virtualization. [Accessed: 23-May-2017].

[3] "Advanced Computer Science (Cloud Computing and Big Data) - MSc - Canterbury," *The University of Kent*. [Online]. Available: /courses/postgraduate/1211/advanced-computer-science-cloud-computing-and-big-data. [Accessed: 23-May-2017].

[4] "Cloud Computing MSc - Postgraduate - Newcastle University." [Online]. Available: http://www.ncl.ac.uk/postgraduate/courses/degrees/cloud-computing-msc/#profile. [Accessed: 23-May-2017].

[5] "MSc in Cloud Computing | NCI." [Online]. Available: https://www.ncirl.ie/Courses/Course-Details/course/MSc-in-Cloud-Computing-MSCCLOUD5. [Accessed: 23-May-2017].

[6] "MSc Cloud Computing :: University of Essex." [Online]. Available: https://www.essex.ac.uk/courses/details.aspx?mastercourse=PG00484&subgroup=1. [Accessed: 23-May-2017].

[7] N. Leavitt, "Is Cloud Computing Really Ready for Prime Time?," *Computer*, vol. 42, no. 1, pp. 15–20, Jan. 2009.

[8] "Top 7 most common uses of cloud computing," *Cloud computing news*, 06-Feb-2014. [Online]. Available: https://www.ibm.com/blogs/cloud-computing/2014/02/top-7-most-common-uses-of-cloud-computing/. [Accessed: 17-May-2017].

[9] L.-J. Zhang, "EIC editorial: Introduction to the body of knowledge areas of services computing," *IEEE Trans. Serv. Comput.*, vol. 1, no. 2, pp. 62–74, 2008.

[10] H. P. Breivold and I. Crnkovic, "Cloud Computing education strategies," in *2014 IEEE 27th Conference on Software Engineering Education and Training (CSEE T)*, 2014, pp. 29–38.

[11] C. Gonzalez, C. Border, and T. Oh, "Teaching in Amazon EC2," in *Proceedings of the 14th Annual ACM SIGITE Conference on Information Technology Education*, New York, NY, USA, 2013, pp. 149–150.

[12] J. Cappos, I. Beschastnikh, A. Krishnamurthy, and T. Anderson, "Seattle: A Platform for Educational Cloud Computing," in *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2009, pp. 111–115.

[13] W. Du and R. Wang, "SEED: A Suite of Instructional Laboratories for Computer Security Education," *J Educ Resour Comput*, vol. 8, no. 1, p. 3:1–3:24, Mar. 2008.

[14] L. Xu, D. Huang, and W.-T. Tsai, "V-lab: A Cloud-based Virtual Laboratory Platform for Hands-on Networking Courses," in *Proceedings of the 17th ACM Annual Conference on Innovation and Technology in Computer Science Education*, New York, NY, USA, 2012, pp. 256–261.

[15] X. Wang, G. C. Hembroff, A. B. Cerier, and B. W. Perrault, "Introducing Cloud Computing with a Senior Design Project in Undergraduate Education of Computer System and Network Administration," in *Proceedings of the 2011 Conference on Information Technology Education*, New York, NY, USA, 2011, pp. 177–182.

[16] H. E. Schaffer, S. F. Averitt, M. I. Hoit, A. Peeler, E. D. Sills, and M. A. Vouk, "NCSU's Virtual Computing Lab: A Cloud Computing Solution," *Computer*, vol. 42, no. 7, pp. 94–97, Jul. 2009.

[17] Y. Khmelevsky and V. Voytenko, "Cloud Computing Infrastructure Prototype for University Education and Research," in *Proceedings of the 15th Western Canadian Conference on Computing Education*, New York, NY, USA, 2010, p. 8:1–8:5.

[18] M. I. Tariq, S. Tayyaba, H. Rasheed, and M. W. Ashraf, "Factors influencing the Cloud Computing adoption in Higher Education Institutions of Punjab, Pakistan," in *2017 International Conference on Communication, Computing and Digital Systems (C-CODE)*, 2017, pp. 179–184.

[19] S. A. Mokhtar, A. Al-Sharafi, S. H. S. Ali, and A. Z. Al-Othmani, "Identifying the determinants of cloud computing adoption in higher education institutions," in *2016 International Conference on Information and Communication Technology (ICICTM)*, 2016, pp. 115–119.

[20] S. Ashtari and A. Eydgahi, "Student Perceptions of Cloud Computing Effectiveness in Higher Education," in *2015 IEEE 18th International Conference on Computational Science and Engineering*, 2015, pp. 184–191.

[21] Á. Mitchell and L. Cunningham, "Impact of cloud computing in Ireland's institutes of higher education," in *eChallenges e-2014 Conference Proceedings*, 2014, pp. 1–11.
[22] I. D. M. Ortega and J. A. B. Ricaurte, "Cloud Computing, trend importance and relevance for higher education," in *2014 9th Computing Colombian Conference (9CCC)*, 2014, pp. 90–95.
[23] "Criteria for Accrediting Engineering Programs, 2016 – 2017 | ABET." .
[24] "Program Educational Objectives." [Online]. Available: https://cs.usu.edu/assessment/objectives. [Accessed: 21-Dec-2016].
[25] "Cloud computing," *Wikipedia*. 16-May-2017.
[26] "10 Hot Technology Trends For 2016," *Network Computing*, 15-Jan-2016. [Online]. Available: http://www.networkcomputing.com/storage/10-hot-technology-trends-2016/520323524. [Accessed: 16-May-2017].
[27] "Home - IEEE Cloud Computing." [Online]. Available: http://cloudcomputing.ieee.org/. [Accessed: 16-May-2017].
[28] M. C. Caraman, S. A. Moraru, S. Dan, and C. Grama, "Continuous Disaster Tolerance in the IaaS clouds," in *2012 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, 2012, pp. 1226–1232.
[29] "Computing platform," *Wikipedia*. 12-May-2017.
[30] "Cloud Solutions - Amazon Web Services (AWS)," *Amazon Web Services, Inc.* [Online]. Available: //aws.amazon.com/solutions/. [Accessed: 16-May-2017].
[31] "Amazon Education." [Online]. Available: https://www.amazon.com/gp/feature.html?docId=1000412651. [Accessed: 08-Jun-2017].
[32] "Education Grants - Free Credits for University Computer Science Classes," *Google Cloud Platform*. [Online]. Available: https://cloud.google.com/edu/. [Accessed: 16-May-2017].
[33] "GitHub Student Developer Pack," *GitHub Education* . [Online]. Available: https://education.github.com/pack. [Accessed: 16-May-2017].
[34] jyoshii270000RYPJ, "IBM Cloud Education Wiki : IBM Cloud Education Resource Guide," 20-Oct-2009. [Online]. Available: https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/WebSphere%20Education%20Wiki/page/IBM%20Cloud%20Education%20Resource%20Guide. [Accessed: 16-May-2017].
[35] "Microsoft Azure in Education." [Online]. Available: https://azure.microsoft.com/en-us/community/education/. [Accessed: 16-May-2017].
[36] "Variables in Your Science Fair Project," *Science Buddies*. [Online]. Available: http://www.sciencebuddies.org/science-fair-projects/project_variables.shtml. [Accessed: 08-May-2017].
[37] "Dependent and independent variables," *Wikipedia*. 27-Apr-2017.
[38] P. R. Rosenbaum, *Design of Observational Studies*. New York, NY: Springer New York, 2010.
[39] "Observational study," *Wikipedia*. 02-Feb-2017.
[40] "Observational Studies." [Online]. Available: http://www.stat.wmich.edu/s160/book/node69.html. [Accessed: 04-May-2017].
[41] F. Shull, J. Carver, and G. H. Travassos, "An Empirical Methodology for Introducing Software Processes," in *Proceedings of the 8th European Software Engineering Conference Held Jointly with 9th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, New York, NY, USA, 2001, pp. 288–296.
[42] "Observational Studies and Secondary Data Analyses To Assess Outcomes in Complementary and Integrative Health Care," *NCCIH*, 22-Jun-2012. [Online]. Available: https://nccih.nih.gov/research/blog/observational-secondary. [Accessed: 04-May-2017].
[43] A. Rabkin, C. Reiss, R. Katz, and D. Patterson, "Using clouds for MapReduce measurement assignments," *ACM Trans. Comput. Educ.*, vol. 13, no. 1, pp. 1–18, Jan. 2013.
[44] W. Zhu, "Hands-On Network Programming Projects in the Cloud," in *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2015, pp. 326–331.
[45] W. Sun, V. Katta, K. Krishna, and R. Sekar, "V-NetLab: An Approach for Realizing Logically Isolated Networks for Security Experiments," in *Proceedings of the Conference on Cyber Security Experimentation and Test*, Berkeley, CA, USA, 2008, p. 5:1–5:6.
[46] A. Abou-Zeid and M. A. Taha, "Accreditation process for engineering programs in Saudi Arabia: Challenges and lessons learned," in *Global Engineering Education Conference (EDUCON), 2014 IEEE*, 2014, pp. 1118–1125.
[47] V. R. B.-G. Caldiera and H. D. Rombach, "Goal question metric paradigm," *Encycl. Softw. Eng.*, vol. 1, pp. 528–532, 1994.
[48] A. Gelman, "Struggles with Survey Weighting and Regression Modeling," *Stat. Sci.*, vol. 22, no. 2, pp. 153–164, May 2007.
[49] "Analysis of variance," *Wikipedia*. 07-May-2017.
[50] "Interpret the key results for One-Way ANOVA." [Online]. Available: http://support.minitab.com/en-us/minitab-express/1/help-and-how-to/modeling-statistics/anova/how-to/one-way-anova/interpret-the-results/key-results/. [Accessed: 19-May-2017].
[51] "What a p-Value Tells You about Statistical Data," *dummies*. .
[52] "Sentiment analysis," *Wikipedia*. 10-May-2017.