MOOCs as a Supplement to Classroom Instruction: An Instructor's Perspective

Jason H. Sharp
jsharp@tarleton.edu
Department of Marketing and Computer Information Systems
Tarleton State University
Stephenville, TX 76401, USA

Abstract

Since their grand arrival on the educational landscape in 2012 Massive Open Online Courses (MOOCs) continue to be a matter of debate and discussion in both the popular press and among academicians. Although perhaps not creating the revolution many advocates thought they would it does not appear that MOOCs will going away any time soon. The rising question it seems now becomes what will be the role of MOOCs in general and more specifically in higher education. This paper grew from my own interest in and participation in a C# Programming MOOC. The purpose is to share my thoughts and experiences about the design, content, and assessment of the course as well as compare it to my own C# programming course using those same themes. The paper begins with a brief history of MOOCs, provides an evaluation of the C# MOOC rooted in the literature, compares the similarities and differences between the MOOC and my course and concludes with a discussion of whether or not MOOCs can serve as a supplement to classroom instruction.

Keywords: MOOCs, C# programming, course design, course assessment

1. INTRODUCTION

In 2012 deemed by one writer as the "Year of the MOOC" (Pappano, 2012) more than a few in the popular press espoused the disruptive nature of Massive Open Online Courses (MOOCS) using phrases such as "campus tsunami" (Brooks, 2012), "historic transformation" (Chubb, 2012), and "higher education revolution" (Friedman, 2012). Since that time much has been written in the popular press as well as in academic circles as to whether or not the MOOC has created or not created a revolution (Daniel, 2012).

While many continue to espouse the virtues of MOOCs over traditional face-to-face and even other types of online courses others have declared the MOOC dead (Lewin, 2013). The truth is likely somewhere in between. George (2014) concludes that "MOOCs are not revolutionary, but are simply a delivery system". In a more balanced perspective Longstaff (2014) states that "despite what their champions, sceptics, and doomsayers suggest – Massive Open Online Courses (MOOCs), as they currently appear, are neither revolutionising nor destroying higher education, they are simply the latest point of tension in its cyclical evolution" (p. 164). De Langen and Van Den Bosch (2014) suggest that the notion that MOOCs will replace traditional education is unlikely; however, due to the potential of MOOCs they cannot be ignored by either traditional or Open universities. Finally, Billington and Fronmueller (2013) surmise that it is not a question of whether MOOCs are here to stay but rather what their role in higher education will be in the future. The goal of this paper is to share my own experience participating in a C# MOOC and assess its content, design, and method of assessment and compare and contrast it with my own C# programming course. The paper begins with a brief history of MOOCs, then provides an evaluation of the C# MOOC I completed, concluding with a comparison with the C# course that I currently teach and suggestions on how a MOOC might be used to supplement classroom instruction.

2. BRIEF HISTORY OF MOOCs

According to most accounts the first MOOC (the term purportedly coined by Dave Cormier) entitled "Connectivism and Connective Knowledge" was developed and facilitated by George Siemens and Stephen Downes in 2008. It is reported that 2,200 students registered for the course. The pedagogy was built on the idea of connectivism which emphasizes the creation of knowledge within a shared community and relies heavily on students to seek out or create their own materials and diffuse it with others in the course. In 2011, Sebastian Thrun and Peter Norvig developed and offered a MOOC entitled "Introduction to Artificial Intelligence" derived from their face-to-face course of the same title. This MOOC is considered by some as the beginning of the modern MOOC movement and representative of many of the MOOCs being offered today. Reports indicate that student registration reach 160,000. Because of the philosophical and pedagogical differences underlying MOOCs they have been categorized as cMOOCs, those that are rooted in connectivist thought and xMOOCs which are considered more behaviorists in nature and structured more closely to traditional classroom pedagogy. Daniel (2014) summarizes that "cMOOCs focus on knowledge creation and generation whereas xMOOCs focus on knowledge duplication" (p. 7)

Currently, the most familiar names associated with xMOOCs include Coursera, edX, and Udacity. Coursera was founded as a for-profit company by Daphne Koller and Andrew Ng from Stanford and is now partnered with Princeton and University of Michigan as well as universities in Europe. edX (formerly MITx because it was founded by MIT), a non-profit company, works with other universities including Harvard, University of Texas, Wellesley College, and Georgetown. In 2012 it launched its first MOOC entitled, "Circuits and Electronics". Finally, Udacity was founded by Sebastian Thrun, David Stavens, and Mike Sokolsky, as a for-profit company. It has no university affiliation and currently has moved toward vocational courses for professionals. Although many xMOOCs are geared toward an academic audience, Butin (2014) suggests that xMOOCS "seem most successful for professional and continuing education" (p. 19). While the MOOCs developed by Coursera, edX, and Udacity have their unique differences, the basic structure consists of video lectures with accompanying online quizzes and projects and/or assignments. A discussion board or other social media component is provided to foster communication. Assessment is either automatic, peer-reviewed or a combination.

3. EVALUATION OF THE C# MOOC

After attending a seminar on Online and Blended Learning and reading Daniel's (2012) article describing the differences between cMOOCs and xMOOCs I decided to wade into the world of MOOCs myself. Since I teach primarily in the area of programming I did a keyword search at Class Central (www.classs-central.com) and MOOC List (www.mooc-list.com) using the term "programming" which resulted in multitude of various programming MOOCs (e.g., Android, C++, Java, Python, Scratch). To further narrow my options I search on the keyword "C# programming" which yielded two results: Programming with C# provided by Microsoft via edX and Beginning Game Programming with C# provided by the University of Colorado System via Coursera. My purpose for enrolling in the C# programming course was two-fold: (1) to actually see what a MOOC was and more specifically how it was designed and (2) to see how the MOOC compared to my own C# programming course. In the spirit of other academics to take and evaluate a MOOC in their discipline I too have ventured to do so and to share my experiences and thoughts (e.g., Bali, 2014; Ben-Ari, 2013; Martin, 2012).

On completion of the MOOC a colleague ask if I would be willing to award credit for my C# programming course to an individual who completed the C# MOOC? This question along with my other reasons for enrolling in the MOOC caused me to begin thinking about how a MOOC might be used to supplement traditional college courses. More specifically, it initiated my thinking in terms of how a MOOC on C#, offered free of charge, might be used to supplement the C# course that I teach at my university. Perhaps similar to what Martin (2012) did by incorporated Thrun and Norvig's MOOC on Artificial Intelligence (AI) into his own AI course at the University of Massachusetts-Lowell. In order to address these areas, I have structured this discussion around the following themes related to the MOOC: (1) design, (2) content, and (3) assessment.

Design

Much has been said about the interactive online experience afforded by MOOCs. Because of this, I was very interested in seeing just how a MOOC was designed. Was the design much different from other types of online courses? More specifically was a MOOC designed substantially different from my own online courses using an LMS such as Blackboard? What I found expressed in the literature was a consistent concern over the lack of serious pedagogy in MOOCs (Bali, 2014; Ben-Ari, 2013; Vardi, 2012). Ben-Ari (2013) who reports on his experience with two MOOCs expressed that his "biggest disappointment came from the complete absence in both courses of any pedagogical innovation" and goes on to say "I see no pedagogical different between these courses and the programming course I taught as a teaching assistant over 30 years ago" (p. 60). Bali (2014) suggests, however, that MOOCs cannot be evaluated from a pedagogical standpoint as a whole, but must be examined individually as some MOOCs contain more sound pedagogy than others. Vardi (2012) concludes that "MOOCs are not the answer to our pedagogical shortcomings" (p. 5).

Inflexibility was also cited as a concern in terms of course design. Video demonstrations, text material, and assessments become static once the course is released (Ben-Ari, 2013). It has also been suggested that a primary component missing from the pedagogy of MOOCs is the role of the teacher (Ross, Sinclair, Knox, Bayne, & Macleod, 2014). In the MOOC that I completed it was stated up front that the instructor would not answer emails or grade assignments. All communication between students and teaching assistants was via a discussion board. Additionally, it is important that faculty members who develop MOOCs are equipped with the skills necessary to do so effectively. As is often the case that professors are thrown in the deep end when it comes to creation of online courses. As Bates and Sangra (2011) assert "it is a myth that professors distinguished by their research output are competent to create online courses without help" (p. 12).

The C# programming MOOC that I completed was very much structured like other xMOOCs. Each content module appeared to be designed with software allowing for the creation of interactive web pages (e.g., SoftChalk Create LessonBuilder) with the ability to navigate back and forth between individual pages. The basic design of the MOOC for each content module was as follows: a short video in YouTube format by the instructor introducing the concept for the module, any number of pages consisting of text, tables, and/or figures related to the content, a varying number of videos demonstrating the concepts, and a varying number of self-assessments in the form of multiple-choice, drag-and-drop, individual selection using a drop-down box, and/or multiple selections using check boxes. In most instances each module concluded with a practice exercise and a peer-reviewed assignment.

For communication between teaching assistants and students a discussion board was the primary vehicle although a live two hour session with the staff was provided about midway into the course. With several thousand students enrolled in the MOOC the discussion board was unwieldy and a bit overwhelming. I only rarely used it for that reason. As Ben-Ari (2013) noted the forums were "too active to be worthwhile reading routinely, and searching is often inadequate to locate the pearls from the dross" (p. 60). Another observation of interest was the interaction between students. At times the postings were rather rude and bordered on harassment, at a minimum some were very unprofessional. Bali (2014) reports a similar experience stating "there were instances of tension and even rudeness among students in several courses" (p. 48). In a study conducted by Breslow, Pritchard, DeBoer, Stump, Ho, and Seaton (2013) it was discovered that only 3% of all students participated in the discussion forum. Not necessarily surprising it was found that certificate earners used the discussion forum at a much higher rate, 52%. Butin (2012) asserted "there will be no online MOOC-driven revolution so long as MITx's discussion board continues to be littered with stressed-out students worried about making deadlines and solving too-difficult questions, students complaining about the repetitiveness of the lectures or celebrating their midterm scores" (paragraph 6).

Content

The C# programming MOOC consisted of twelve modules completed over a six-week time period. The syllabus topics are provided in Table 1:

As mentioned earlier, the content was delivered via short video lectures by the instructor as well as static web pages containing text, figures, and tables. It was possible to download the video lectures and video transcripts for future reference offline. At the end of each module an assessment in the form of true-false, drag-and-drop, or check boxes, or multiple-selection was provided. These assessments were required and were auto-graded. For the most part each module contained a practice exercise and a required assignment which was peer-graded. Overall, the content was easy to navigate and had a logical flow. The text portions were broken up into multiple pages so that not too much scrolling down the page was necessary and it kept from making the text seem overwhelming. The course included additional resources in the form of web links to various Microsoft web site pages as well as an accompanying textbook, Sharp (2013). However, the textbook was not required for the course.

C# MOOC Course Syllabus
Module 1 - Introducing C#, The Tools, Data Types, Variables, Operators, and Expressions
Module 2 - Decision and Repetition Statements
Module 3 - Methods and Exception Handling
Module 4 - Working Arrays, Enumerations, and Structures
Module 5 – Classes, Encapsulation, Static Methods and Static Classes, and Anonymous Classes
Module 6 – More OOP and Resource Management (Garbage Collection)
Module 7 – Collections
Module 8 - Working with Generics
Module 9 – Using Events and Delegates
Module 10 – Using LINQ
Module 11 – Multitasking and Parallel Programming
Module 12 – Asynchronous Programming

Table 1

As mentioned earlier, the content was delivered via short video lectures by the instructor as well as static web pages containing text, figures, and tables. It was possible to download the video lectures and video transcripts for future reference offline. At the end of each module an assessment in the form of true-false, drag-and-drop, or check boxes, or multiple-selection was provided. These assessments were required and were auto-graded. For the most part each module contained a practice exercise and a required assignment which was peer-graded. Overall, the content was easy to navigate and had a logical flow. The text portions were broken up into multiple pages so that not too much scrolling down the page was necessary and it kept from making the text seem overwhelming. The course included additional resources in the form of web links to various Microsoft web site pages as well as an accompanying textbook, Sharp (2013). However, the textbook was not required for the course.

Assessment

Assessment is crucial to the success of MOOCs and may present a major obstacle. Wherever MOOCs are discussed this topic finds itself easily in the list of top challenges. The questions arises "just how do instructors effectively evaluate students in such a massive environment"? As Hyman (2012) states, "it's easy when the tests are multiple choice or when the work is peer-graded, but technology still needs to catch up to evaluating the rest" (p. 21). Some, however, would question the ease at which peer grading can be implemented and to its effectiveness. It does not take long to realize that there are various perspectives on this issue (e.g., McEwen, 2013; Morrison, 2013, Nedlinger, 2013; Rees, 2013; Watters, 2012). While many of the debates regarding peer assessment take place in the popular press or in opinion pieces Luo, Robinson, and Park (2014) assert that while research appears to support the legitimacy of peer grading within traditional courses the ability to apply the same approaches to MOOCs is in need of additional research. As such they conducted a study to examine the reliability, validity, and perceived effects of peer grading in a MOOC entitled Maps and the Geospatial Revolution (www.coursera.org/course/maps). Their findings suggest that peer-grading approaches can provide reliable and valid results as well as a positive experience for the students. In fact over 60% of students reported that the grades they received were fair and the feedback received from their peers was useful.

Adding to the challenge of assessment in a MOOC is the issue of whether the student can receive credit or a certificate, often designated as "honor code" or "verified" (Ben-Ari, 2013; Hyman 2012). Much has been written in the popular press about universities awarding credit hours for the successful completion of a MOOC. Two examples of well-known universities doing just that have made the headlines. Georgia Tech has partnered with Udacity and AT&T to offer a Master of Science degree in computer science completely online via MOOCs (Georgia Tech, 2013) while Arizona State University and edX released an announcement for the Freshman Global Academy which will provide courses for completion of the entire freshman year via MOOCs for a registration fee of $45 and the option to pay to receive credit at the end of the course (Straumsheim, 2015). On the professional front, it has been suggested that if a student has good grades on their academic record and has received a certificate that some companies will be willing to include that certificate in the assessment of the potential employee in a favorable light (Hyman, 2012).

The course consisted of two types of assessment: self-assessment and peer-review assignments. As mentioned under design of the course the self-assessments were distributed throughout module and consisted of multiple-choice, drag-and-drop, individual selection using a drop-down box, and/or multiple selections using check boxes. The programming assignments were peer-reviewed by two or three other students in the course as follows: 5 points – good, 3 points – fair, 1 point – poor.

The peer-review of the programming assignments was quite a source of contention in the course. The participants ranged from those without any prior programming experience to professional programmers with many years of experience. From a reading of the discussion posts it was apparent that there were issues with novice programmers reviewing assignments submitted by more experienced programmers. This issue was even more complicated by the fact that the more experienced programmers would write programs which far exceeded the assignment requirements. Throughout the course, the instructor and/ or teaching assistant (TA) had to address this issue and by the end of the course a very detailed explanation of how to peer-review assignments was provided. When all else failed the TA simply ask if a student did not understand the code which was assigned to them for peer-review that they simply skip it and move to the next one assigned to them.

To receive a certificate of achievement for the MOOC a student must receive 70% on self-assessments and peer-reviewed assignments. The certificate was of two types: honor code and verified. The honor code certificate of achievement certifies that the student successfully completed the course, but did not verify their identity. Honor code certificates are currently free. A verified certificate of achievement certifies that the student successfully completed the course and verified their identity through a photo and picture ID. The fee for the verified certificate for this course was $90.00.

4. COMPARISON OF THE C# MOOC TO MY C# PROGRAMMING COURSE

A stated purpose of this paper is to compare the design, content, and assessment of the C# MOOC to the C# programming course that I teach at my university. This course is taught both face-to-face and online and averages around 20-30 students in each section.

Design

As noted above when I began to hear about MOOCs I was very curious about how their design was similar or different from the courses that I teach. At my university we currently use Blackboard Learn 9 as our learning management (LMS) and I have designed my C# programming course using the tools provided in Blackboard. In terms of the design of the MOOC and my course there are quite a few similarities. I have created a large repository of video demonstrations which students may view at their convenience and as many times as they wish. One difference is that currently students are not able to download and view them outside of Blackboard. I am currently working to create a YouTube channel so that I can post the videos in that format as I receive request from students to have availability to the videos even after the course has been completed. I also require students to take quizzes which are auto-graded, but unlike the MOOC my quizzes are not embedded into web pages along with the videos, they are placed under a separate link entitled, Chapter Quizzes. I have considered creating SoftChalk presentations to emulate the design which I experienced in the MOOC.

I must admit that my course suffers from the inflexibility that is often associated with MOOCs in that after the content is posted it becomes more or less static. Considering that I am very involved with my students online, however, I am able to make adjustments along the way which would be difficult for the builder of the MOOC which involves so many more students. Through my involvement not only am I able to provide a more dynamic learning environment than the MOOC I completed I am also available on a regular basis to my students via email, phone, and a class discussion board. As I noted earlier in the MOOC it was explicitly stated up front that the instructor would not answer email questions or grade assignments. Basically the role of the instructor of the course was to record the video demonstrations.

Communication in my course is conducted via email, chat sessions, phone calls, as well as a discussion board for general questions. This is different from the MOOC which utilized only a discussion board for communication between teaching assistants and students. Again, with thousands of students enrolled in the course the discussion board for me was not very effective. It was too time intensive for me to keep up with all of the postings. I focused primarily on the post by the teaching assistants. Fortunately, these posts were pinned to the top of the discussion board and were easy to access.

Overall, I thought the design and pedagogical approach of the MOOC was sound. However, I did not find it significantly different than the course that I have created in Blackboard. It consists of all the same components (videos, quizzes, assignments, exams, discussion board, email, etc.) just in a different format. I realize that I have only completed one MOOC provided by one entity, so on participating in other MOOCs I may encounter the cutting-edge technology that we so often hear about in MOOCs.

Content
In comparison to the MOOC my C# programming course is broken into two 16-week semesters with the first semester being the introductory course while the second semester is the advanced course. Two exams plus a final exam are typically given in week 6, 13, and 16. The course meets two days a week for an hour and fifteen minutes.

Overall, the MOOC and my course cover the majority of the same content. Both include an introduction to C# and working in the Visual Studio environment. Basic topics including data types, variables, arithmetic operators, logical operators, calculations, named constants, and exception handling are all covered. The three logic structures: sequence, selection, and repetition are included. Both courses cover exception handling, arrays, collections, methods and events. Some of the content related to object-oriented programming in the MOOC is covered in the 2nd semester of my course. The MOOC did cover some additional material not included in my course including: generics, multitasking and parallel programming, and asynchronous programming. While my course does cover basic database theory, design, and implementation it does not specifically cover LINQ. A comparison of the content of the two courses by module/chapter is provided in Appendix A.

Due to the differences in the time frame for the MOOC as opposed to my course, the MOOC covers the content much more quickly. I found that I had to stay on top of the MOOC on almost a daily basis in order not to get behind. Students in my course have much more time to digest the material, but I'm sure many still wait till the last minute start working on assignments and/or studying for exams.

A noticeable difference between the MOOC and my own course is that programming examples and assignments in the MOOC were primarily console-based with a very minimal exposure to GUI applications. My course begins with GUI development from day one and continues for the remainder of the course. The debate about whether programming, in particular the first course, should be taught using a console-based approach as opposed to a GUI-based approach is not new (i.e., Bishop-Clark, 1998) and I have debated it myself. A further discussion, however, is perhaps beyond the scope of this particular paper. I note it here because this is a significant difference between the two courses.

Assessment
In terms of assessment the MOOC and my course were similar in that quizzes and assignments make up a large portion of the grade. In both cases quizzes are auto-graded, but a notable difference is in the assessment of the assignments. In the MOOC all assignments are peer-graded while in my course I grade each assignment based upon a detailed grading criteria. Also, in my course exams are required for all the chapters covered, also graded by me. Based upon my personal experience in the MOOC peer-grading potentially causes a much more significant problem than described by Hyman (2012). In the MOOC there was a good amount of consternation among students posted on the discussion board about the fairness of the peer assessment system. This issue rose to a level to occasion the teaching assistant to finally provide a very detailed set of instructions regarding how the peer grading process should be done. Unfortunately, this came several weeks into the course after a good deal of damage had been done regarding the quality of the course. The issue of appealing a peer-assessment was posted regularly, but there was no process in place to address this matter.

5. LIMITATIONS AND FUTURE RESEARCH

An obvious limitation of this paper is that it is only analyzing one MOOC provided by one entity. To arrive at a broader understanding of the design, content, and assessment of MOOCs it would be necessary to review other MOOCs offered by other entities. Another limitation is the simple fact that this paper is the experience of a single person comparing the MOOC with his own course. Subjectivity, bias, and personal preference can be minimized, but cannot be completely removed from the process. This is much more of an experience report grounded in the literature than a rigorously designed empirical study.

As this paper recounts my own experiences with a C# MOOC and how it compares to my own C# course in terms of content, design, and assessment many questions remain for future research such as: Should MOOCs have prerequisites similar to traditional college courses? Does learning style or maturity have an effect on the types of students who would do well in a MOOC and how does this compare to traditional and online courses? How would MOOCs be integrated into a traditional college course and would the MOOC be supplemental or required? These and many other questions appear to be ripe for future study.

6. CONCLUSION

Overall, my experience participating in the C# MOOC was interesting and positive. In terms of the course design I found that there were some aspects that I really liked and perhaps will be able to employ such as creating SoftChalk presentations to embed video, text, and assessments into an easily navigable lesson or module. I also liked having the video demonstrations in YouTube format and being able to download them to watch offline.

In relation to course content I felt for the most part that the combination of my two semester sequence covered the large majority of material covered in the MOOC, albeit at a much slower pace. I am also amenable to considering the additional content provided in the MOOC for future iterations of my course. My personal feeling is that students seem to show more interest and responsiveness when GUI applications are developed in a programming course although from a pedagogical perspective I see the potential benefits of first learning to code from the console. I think students like to feel like they are programming applications like the ones they work with on a daily basis rather than text in a black box. Even more so in the future perhaps web-based development will replace Windows application development even in the introductory course.

Finally, in regard to course assessment I have no real issues with quizzes being auto-graded and providing immediate feedback to the student. My rationale for quizzes is to get the students into the material in hopes they will begin to utilize the textbook and other course resources provided. At least in this specific MOOC the peer-assessment process was a concern and certainly is an area of improvement for the creator of the MOOC. With such a large, diverse group of students the peer-grading was a big challenge. I do know from reading the literature that some instructors do incorporate peer-grading into their courses and I do see a benefit to it. But it is typically on a much smaller scale and the instructor has much more of an ability to monitor it.

In answer to my original question, I do feel like the C# MOOC would be beneficial as a supplement to the students in my own C# programming course. While the content is very similar, the design and the environment are different enough to provide the students with a different perspective than my own and the required assignments were sufficiently challenging. In terms of specific ways a MOOC might be incorporated as a supplement to classroom instruction certain topics that are difficult for students might be assigned from the MOOC in addition to classroom instruction. Again, providing different approaches to the same topic might be beneficial to the students. Using a MOOC as the starting point for discussion similar to what Martin (2012) did with his AI course is another possible direction.

To the question of whether or not I would award credit to a student who received a certificate in the course I think I am still on the fence. The issue is much like what we face in our own online courses that of ensuring that students are doing their own work as opposed to someone else doing it for them.

7. REFERENCES

Baggaley, J. (2014). MOOCS: Digesting the facts. *Distance Education*, 35(2), 159-163.

Bali, M. (2014). MOOC pedagogy: Gleaning good practice from existing MOOCs. *MERLOT Journal of Online Learning and Teaching*, 10(1), 44-55.

Bates, A. W., & Sangra, A. (2011). Managing technology in higher education: Strategies for transforming teaching and learning. Jossey-Bass: San Francisco, CA.

Ben-Ari, M. (2013). MOOCs on introductory programming: A travelogue. *ACM Inroads*, 4(2), 58-61.

Billington, P. J., & Fronmueller, M. P. (2013). MOOCs and the future of higher education. *Journal of Higher Education Theory and Practice*, 13(3/4), 36-41.

Breslow, L., Pritchard, D. E., DeBoer, J., Stump, G. S., Ho, A. D., & Seaton, D. T. (2013). Studying learning in the worldwide classroom: Research into edX's first MOOC. *Research & Practice in Assessment*, 8, 13-25.

Butin, D. W. (2012). What MIT should have done. *eLearn*, 6, Retrieved from http://elearnmag.acm.org/archive.cfm?aid=2263018

Brooks, D. (2012). The campus tsunami. *The New York Times*, May 3, Retrieved from http://www.nytimes.com/2012/05/04/opinion/brooks-the-campus-tsunami.html?_r=0

Chubb, J. E., & Moe, T. M. (2012). Higher Education's Online Revolution. *The Wall Street Journal*, May 30, Retrieved from http://www.wsj.com/articles/SB10001424052702304019404577416631206583286

Daniel, J. (2012). Making sense of MOOCs: Musings in a maze of myth, paradox, and possibility. *Journal of Interactive Media in Education*, 18.

De Langen, F., & Van Den Bosch, H. (2014). Massive open online courses: Disruptive innovations or disturbing inventions? *Open Learning*, 28(3), 216- 226.

Dolan, V. L. B. (2014). Massive online obsessive compulsion: What are they saying out there about the latest phenomenon in higher education?

Friedman, T. L. (2012). Come the revolution. *The New York Times*, May 15, Retrieved from http://www.nytimes.com/2012/05/16/opinion/friedman-come-the-revolution.html?_r=1

Georgia Tech (2013). Georgia Tech announces massive online master's degree in computer science. Retrieved from http://www.news.gatech.edu/2013/05/14/georgia-tech-announces-massive-online-masters-degree-computer-science

George, J. (2014). The future of higher education in the U.S.: MOOCs, a case in point. *AIS Success Strategies Webinar Series*.

Guzdial, M. (2013). Results from the first-year course MOOCs: Not there yet. *Communications of the ACM*, 57(1), 18-19, Retrieved from http://cacm.acm.org/blogs/blog-cacm/168862-results-from-the-first-year-course-moocs-not-there-yet/fulltext

Hyman, P. (2012). In the year of disruptive education. *Communications of the ACM*, 55(12), 20-22.

Lewin, T. (2013). After setbacks, online courses are rethought. *The New York Times*, Retrieved from http://www.nytimes.com/2013/12/11/us/after-setbacks-online-courses-are-rethought.html?_r=0

Luo, H., Robinson, A. C., & Park, J. (2014). Peer grading in a MOOC: Reliability, validity, and perceived effects. *Journal of Asynchronous Learning Networks*, 18(2), 1-14.

Martin, F. G. (2012). Will Massive Open Online Courses Change How We Teach? *Communications of the ACM*, 55(8), 26-28.

McEwen, K. (2013). Getting to know Coursera: Peer assessments. Retrieved from http://cft.vanderbilt.edu/2013/01/getting-to-know-coursera-peer-assessments/

Morrison, D. (2013). Why and when peer grading is effective for open and online learning. Retrieved from https://onlinelearninginsights.wordpress.com/2013/03/09/why-and-when-peer-grading-is-effective-for-open-and-online-learning/

Pappano, L. (2012). The year of the MOOC. *The New York Times*, Retrieved from http://www.nytimes.com/2012/11/04/education/edlife/massive-open-online-courses-are-multiplying-at-a-rapid-pace.html?_r=0

Neidlinger, J. (2013). Doss peer grading of essays really work in a Coursera online class? Retrieved from http://loneprairie.net/peer-grading-coursera/

Rees, J. (2013). Peer grading can’t work. Retrieved from https://www.insidehighered.com/views/2013/03/05/essays-flaws-peer-grading-moocs#ixzz2MiKxNP7b

Ross, J., Sinclair, C., Knox, J., Bayne, S., & Macleod, H. (2014). Teacher experiences and academic identity: The missing components of MOOC pedagogy. *MERLOT Journal of Online Learning and Teaching*, 10(1), 57-59.

Sharp. J. (2013). Microsoft Visual C# 2013 Step By Step. Microsoft: Seattle, WA.

Straumsheim, C. (2015). MOOCs for (a Year’s) Credit. *Inside Higher Ed*, Retrieved from https://www.insidehighered.com/news/2015/04/23/arizona-state-edx-team-offer-freshman-year-online-through-moocs

Vardi, M. Y. (2013). Will MOOCs destroy academia? *Communications of the ACM*, 55(11), 5.

Appendix A

C# Programming MOOC	My C# Programming Course
Module 1 - Introducing C#, The Tools, Data Types, Variables, Operators, and Expressions	Chapter 1 – Software Development Process and Working with Visual Studio (Week 1-2) Chapter 2 - Forms, Simple C# Code, Label and PictureBox Controls, Comments (Week 3-4) Chapter 3 – TextBox Controls, Variables, Data Types, Operators, Calculations, Input/Output of Numeric Values, ToString Method, Exception Handling, Named Constants, Fields (Week 4-6)
Module 2 - Decision and Repetition Statements	Chapter 4 – If Statements, Logical Operators, Boolean Values, Flags, String Comparisons, TryParse Method, Input Validation, Radio Buttons, Check Boxes, switch Statement, List Boxes (Week 6-7) Chapter 5 – List Boxes, while, for, do Loops, Increment operators (Week 9-10)
Module 3 - Methods and Exception Handling	Chapter 3 – Exception Handling (Week 4-6) Chapter 6 – Methods, Passing Arguments By Value, Reference, Name, Output Parameter, Void and Value-Returning Methods, Boolean Method (Week 10-11)
Module 4 - Working Arrays, Enumerations, and Structures	Chapter 7 – Arrays and Lists (Week 12-13)
Module 5 – Classes, Encapsulation, Static Methods and Static Classes, and Anonymous Classes	Chapter 9 – Classes and Multiform Projects (2nd Semester – Week 2-3)
Module 6 – More OOP and Resource Management (Garbage Collection)	Chapter 10 – Inheritance and Polymorphism (2nd Semester 3–4)
Module 7 – Collections	Chapter 7 – Arrays and Lists (Week 12-13)
Module 8 - Working with Generics	Not covered
Module 9 – Using Events and Delegates (first exposure to GUI applications)	All chapters since GUI applications were developed from start of the semester.
Module 10 – Using LINQ	Chapter 11 – Databases (LINQ not specifically covered)
Module 11 – Multitasking and Parallel Programming	Not covered
Module 12 – Asynchronous Programming	Not covered