

Evaluation of Pace University's Master's Degree Program for BNY Mellon Employees

Scott Cruzan, Raymond K Ho, Inna Kolbovskaya, Lalji Paul, Russell Petersen and Sotiris Skevoulis

Pace University Seidenberg School of CSIS, White Plains, NY 10606, USA

Abstract

This paper is the result of research performed by a class of Bank of New York Mellon Information Technology Professionals who are completing their master's degrees in Software Development and Engineering at Pace University through a program developed jointly by Pace University and the BNY Mellon Information Technology department. Our assigned task involved documenting our experiences during three and a half years of graduate level instruction by members of the Pace University faculty. These courses were in some cases delivered on-site at BNY Mellon facilities with off-site participants and were in other cases delivered entirely online with conference calls and online presentation sharing. Evaluations of the strengths and weaknesses of each completed course were documented along with comparisons to similar graduate level courses taught at other universities. Also, comments were gathered relating to the applicability of each course and the overall curriculum to our roles in the BNY Mellon IT department.

1 Introduction

In September of 2008 BNY Mellon's IT Learning & Development group announced the Advanced IT Graduate Certificate Program offered through Pace University. The courses were graduate level courses applicable to Pace University Master's degree programs. They were taught by instructors from Pace University. Classes were to be held one night a week at the BNY Mellon office of 101 Barclay Street, in downtown Manhattan (about 4 blocks from the Pace downtown campus) and over the internet / conference call bridge, for students who could not attend on site. We were offered the choice between two programs of study: Software Development & Engineering or Security & Information Assurance. Each program was comprised of 5 courses. The courses for the certificate began in January 2009 and completed in June 2010. We were also informed that upon completion of the certificate we could continue our education through Pace to pursue our Master's degree. Of the students who have previously gone through the program, 70% to 80% had continued on for their master's degree.

The courses offered ended up being slightly different from what was initially presented. For the Security program we took Software Risk Management instead of Building Secure Software II. This was appropriate for the objectives of the certificate; Software Risk Management was an excellent and challenging course. In the software engineering track Requirements Engineering was offered instead of Software Engineering Seminar I. This made sense since requirements engineering is a critical early step in software engineering. Students on this track still took Software Engineering Seminar II without previously having taken Software Engineering Seminar I. The professor tailored the course accordingly.

After we completed our graduate certificates about half of us decided to continue for our master's degrees. Because of the reduced number of people continuing for the master's degree, it was no longer feasible to have 2 independent programs. It was announced that the programs would be merged into a single program: a master's degree in Secure Software Development &

Engineering. Although the separate tracks to more specialized master's degrees were eliminated, we appreciated the opportunity to continue taking classes for a master's degree. For the master's program on site classes at 101 Barclay were no longer offered, the classes were to be only virtual, given online with a conference call. Although attending class in person provides a better learning experience than a virtual class, the onsite classes were often not well attended. People who worked close to, or even in 101 Barclay would often still attend the class virtually, so it is understandable that the on-site class was eliminated.

It was announced that Pace would allow students to earn 6 credits toward their master's degree by passing the IEEE Certified Software Development Associate (CSDA) exam, demonstrating understanding of the Software Engineering Body of Knowledge (SWEBOK.) Students were informed later that passing the exam would be worth only 3 credits. There were also changes in the courses that were offered. Software Development Studio I and II were replaced by Software Research Seminar and Computer Science Projects. Also, the degree that will be earned is a master's degree in Software Development & Engineering, not Secure Software Development & Engineering as was previously indicated. Students may have been less frustrated at learning that another course would be required if it had been initially stated that the program was subject to change.

Following is a critique of the courses we took, including comparisons with comparable courses at other universities. Since we represent students from both tracks, not all of us took the same classes for the first half of the program.

2 Goals

To evaluate the following programs at Pace University tailored for employees of BNY Mellon

- Advanced IT Graduate Certificate Program in Software Development & Engineering
- Advanced IT Graduate Certificate Program in Security & Information Assurance
- Master's degree program in Software Development & Engineering

Programs are evaluated by the experience of students who participated in the program and comparing the courses with comparable courses in other universities

3 Methodology

Each class was assigned a primary reviewer, who reviewed the class and compared it with comparable classes from other universities. The other students who took the class each reviewed the document and added comments as appropriate.

4 Details

The courses we took with a brief synopsis are as follows:

4.1 Advanced IT Graduate Certificate Program in Security & Information Assurance courses

IT-603 Overview of Information Security

This course covered the Security Framework, Access Control & Site Security, TCP/IP for Network Security, Subnetting & Attack Methods, Computer Firewalls, Host Security, Cryptography, SSL/TLS, VPNs, Application Security, Incident & Disaster Response, Managing the Security Functions, Cyber Laws & Cyberwar. There were 10 timed quizzes, 6 homework assignments and a team project. In comparing this course with Binghamton University - CS458/CS558: Introduction to Computer Security we found that CS458/CS558 included an Introduction to network and socket programming, Security policies and principles and Advanced topics (database security, virtual machine security), required software gcc, g++, Java, and

Openssl. When we compared it to Lehigh University – CSE 397/497 Software System Security we found that CSE 397/497 surveys common software vulnerabilities, including buffer overflows, format string attacks, cross-site scripting, and botnets. It also discusses common defense mechanisms, including static code analysis, reference monitors, language-based security, secure information flow, and others. Subjects covered in the Pace course that were not covered in the other courses were Incident and Disaster Response and Cyber Laws and Cyber War. Despite these differences we found for the most part the course to be in line with the similar courses in other universities.

IT-660 Network Security

This course covered Risk Analysis, Traffic Signatures & Intrusion Detection Systems, Firewall Configurations, Virtual Private Networks, DNS Security and Advanced Security Topics. There were 4 timed quizzes, 5 homework assignment labs, and a team project. In comparing this course with Lehigh University – CSE 343/443 Network Security, CSE 343/443 also covered Worm modeling and detection and Wireless security Issues. In comparing it with Columbia University – COMS W4180 Network Security, COMS W4180 had the advantage of covering foundations of network security and an in-depth review of commonly-used security mechanisms and techniques, security threats and network-based attacks, applications of cryptography, authentication, access control, intrusion detection and response, security protocols (IPsec, SSL, Kerberos), denial of service, viruses and worms, software vulnerabilities, web security, wireless security, and privacy. The disadvantage of the Columbia course is that it required that you must be able to program in C or C++ and you should have a decent knowledge of networks. Networking (W4119) is a prerequisite. Subjects covered in the Pace course that were not covered in the other courses were Risk Analysis, security policy, DNS Security, Virtualization and Cloud computing. For the most part we found this course is in line with similar courses in other universities, more programming assignments may be required by other schools as well as networking knowledge.

IT-666 Information Security Management

This course covered information security from organizational and managerial perspectives. Topics covered were Introduction to Management of Information Security, Planning for Security, Planning for Contingencies, Security Policy, Developing the Security Program, Security Management Models and Practices, Risk Management: Identifying and Assessing Risk, Risk Management: Assessing and Controlling Risk, Protection Mechanisms, Personnel and Security, Law and Ethics, and Information Security Project Management. There were 4 homework assignments, one of which was a semester long project, 9 team presentations (3 per team) and a team project. In comparing this course with James Madison University's MS Computer Science with Concentration in Information Security we found that the program has in depth study of forensics and James Madison University is the national center of excellence for Information Security. However the courses do not cover a management or senior leader's perspective of the Information Security Organization. In comparing this course with Yale University, School of Computer Science, CPSC 567bu, Cryptography and Computer Security we found the Yale course to be very technical, lot of technical contents are covered in one course. However the Yale course does not give the management framework. Advantages of the Pace course is that it helps the students to master the Information Security Management Leadership needs and since Information Security is a new and growing field. Many organizations are going through the process of setting up the Information Security Organization. This course gives the students the material to master the steps and processes to setup the organization. The disadvantage is the course may not be that relevant for an information security professional who has an emphasis on the technical aspects of Information Security. We found the Pace course gives very robust management framework to the Information Security program. The other courses are either very technical for an expert information Security Practitioner or entry level courses that give awareness to the program.

CS-870 Software Risk Management

This course covered Software Security Touchpoints (i.e., software security best practices), including: Code Review, Architectural Risk Analysis, Penetration Testing, Risk-Based Security Testing, Abuse Cases, Security Requirements, and Security Operations. Additional topics such as Enterprise Software Security Program were also covered.

This course also served as a preparation course for CISSP certification exam and provided a review of the CISSP - Common Body of Knowledge (CBK), including 1) Access Control; 2) Application Development; 3) Security, Business Continuity & Disaster Recovery Planning; 4) Cryptography; 5) Information Security Governance & Risk Management; 6) Legal, Regulations, Investigations & Compliance; 7) Operations Security; 8) Physical (Environmental) Security; 9) Security Architecture & Design; 10) Telecommunications and Network Security.

There was a one team project, ten online quizzes and a team research project. The course topics were at the graduate level standards and important for an Information Technology professional. The Risk Management Framework including the Risk Identification, Risk measurement, Risk Monitoring, Risk Mitigation and Risk Management were useful subjects. This course opened up to multifaceted use of Risk management in daily work life. This course put a special emphasis on the Risk Management of Software components and Technology components.

This course has an in-depth review of the Secure Software Development. The Professor gave us the insight on the best practices of Secure Software development best practices. The course served as a foundation for the CISSP Certification. The Pace university program provided a more comprehensive coverage of Risk Management when compared to the relevant courses at the Boston University, MS Business continuity, Security and Risk Management.

IT-668 Building Secure Software

This course covered an introduction to expert perspectives and techniques that will help them to ensure the security of essential software. The course also covered the process to consider threats and vulnerabilities early in the development cycle to build security into their software systems and determination an acceptable level of risk, development of security tests, and techniques to plug security holes. [1]

The topics included Engineering Secure Software, Technical trends, security goals, common security pitfalls, The Rise of Security Engineering; The role of software security personnel, common criteria, software risk management in practice; Security requirements engineering; Comparing Open Source and Closed Source; Secure Software Design, Securing the weakest link, the Principle of Least Privilege, practicing defense in Depth, SecureUML; Architectural Security Analysis, , implementation of security, using RATS in analysis and Access Control & Password Authentication.

The students had a project that addressed the Secure Software Development Life Cycle. A deeper dive into Secure Software Development Life Cycle with special emphasis on the needs of application development and Operating systems and middleware development would have given the right depth to the course.

The Pace University course is a good course on Building Secure Software. However, the George Mason University courses may have more depth in the building of secure software.

4.2 Advanced IT Graduate Certificate Program in Software Development & Engineering courses

CS-775 Requirements Engineering

In this course students were presented with theory and techniques related to Requirements Engineering. Students were divided into groups and charged with documenting

requirements for a software development project of their choosing allowing them to apply the techniques that were covered by the lectures. Groups were constructed with at least one remote student in each group giving each group the same challenge of collaborating remotely. In addition to the project, students were evaluated via 2 exams.

This course was compared with similar offerings at University of Texas at Dallas and University of Ottawa and no significant differences in the course content were found. Each of these programs included a group project involving gathering and documenting software requirements.

CS-773 Software Design Methodologies

In this course students were instructed in topics and techniques related to software design. Each student applied these techniques in an individual design project. These projects were based on Object-Oriented design techniques. An emphasis was placed on understanding and applying standard Object-Oriented design patterns. Each student's project was required to incorporate at least one design pattern.

One of the best features of this course was that students presented their designs both in development mid-course and at the end of the term. This allowed each of us to learn from the experiences of the others.

This course was compared with a similar one offered at the University of Nebraska – Lincoln. The courses seemed to be very similar with the notable exception that the University of Nebraska course allowed students to complete group (instead of individual) requirements project or complete an individual research project on a Software Design topic. This could be seen as either an advantage or a disadvantage depending on the goals of the individual student.

CS-616 Software Engineering Seminar II

This course covered and reviewed software engineering topics with a hand-on software application project. Students were expected to design, implement and demonstrate a large software project using an object oriented programming language such as Java or C#. Students worked in groups of three or four. Each group set up a wiki for project files depository and communication use.

Students were required to complete the assigned work including documentation, design, implementation, application delivery and demonstration. For each milestone of the project, every group of students was expected to provide the status of the project and the required material to the instructor. After reviewing the submitted material, the instructor provided feedback and helped students make any necessary adjustment their projects. The project assignment simulates the real world project work flow. So after the class, students will have solid foundation to deal with real projects on the job

CS-865 Distributed Systems

This course covered design and implementation of network distributed systems, design and formal specification of client-server applications using state machines and communication protocol techniques. Students also learned the fundamentals and techniques of developing distributed object-oriented applications, using a patterns-based approach. [2]

The course topics included: Characterization of Distributed Systems, System models, Networking and Internetworking, Interprocess Communication, Distributed Objects and Remote Invocation, Operating System support, Security, Distributed File Systems, Naming Services, Peer-to-Peer Systems, Time and Global States, Coordination and Agreement, Transactions and Concurrency Control, Distributed Transactions, Replication, Mobile and Ubiquitous Computing, Distributed Multimedia Systems, Distributed Shared Memory, Web Services

The course required every three or four students to form a team for a team project. There were four homework assignments and two team-led presentations and the final term project.

CS-825 Component Architectures for Enterprise Application Development

This course covered the following topics: introduction, overview of the principles and architecture of distributed computing, the concepts, frameworks, and architecture of Enterprise JavaBeans (EJB), Resource management, EJB basics, simple entity and session beans, and EJB clients, EJB tools and deployment, Entity beans and basic persistence, EJB-QL, Bean-managed persistence, Session beans, security Project presentation. The course focused on object-oriented component architectures for enterprise applications. We attended online lectures weekly. We were required to complete a final exam and a research project at the end of the course.

4.3 Master's degree in Software Development & Engineering courses

CS-615 Software Engineering Seminar I

This was a graduate-level course encompassing the full range of Software Engineering activities. In this course we learned through lectures, reading assignments, and group software projects. The focus was on Object Oriented design.

The group software projects were constructed using the Tersus (www.tersus.com) language which uses diagrams representing the software to be compiled into an executable web application. This was selected by the professor because many class members were not programmers and would not be comfortable programming in Java or any similar language.

Students found the course content to be very useful, but they were frustrated by the use of Tersus since it was not completely documented and much experimentation was needed to get the intended results.

CS-777 Software Reliability and Quality Assurance

This course covered Quality management: defining standards and making sure standards are followed, Verification and Validation: Assuring that a software system meets a user's needs, Critical Systems Validation: Validating the reliability, safety and security of computer-based systems, Configuration management: Managing the products of system change, Defect testing: Testing programs to establish the presence of system defects, Software change: Managing the processes of software system change, Dependable software development: Programming techniques for building dependable software systems. There was a midterm exam, four team presentations (one per team) on software quality papers chosen by the professor, and a team term project. For the term project, teams would periodically present their project status to the class. In comparing this course with Stevens Institute of Technology - SSE 567 Software Testing, Quality, and Maintenance we found that the Stevens course had a greater coverage of traceability matrices, however it did not have a full term class project. Instead it had weekly projects, but there is likely more value in a semester long project where the quality of all aspects of a software project is evaluated. In comparing this course with University of Alabama- CS 491/591 Software Testing and Quality Assurance we found it comparable in coverage of topics to the Pace course, however it did not have a class project comparable to the Pace course. The graduate level course CS 591 includes a term paper, but it only counts for 20% of the grade. We found many universities with otherwise strong computer science or software engineering programs did not have a separate course in software quality assurance. Those that did have a software quality assurance course did not have a major class project.

SE-679 Contemporary Software Engineering Theory and Practice

This course is an introduction to advanced software engineering research and methodologies. It surveys the entire field of software engineering, drawing upon the latest advances that have been published in software engineering journals and conference

proceedings. Topics covered included Software Engineering Ethics, Requirements Engineering, Software Design, Software Testing, Software Maintenance, Software Configuration. [3]

This course served as a preparation to CSDA certification exam and provided a review of the SWEBOK – Software Engineering Body of Knowledge. Weekly lectures were tied to the SWEBOK knowledge areas and on-line quizzes were helpful in our understanding of the concepts. We had a final open-book exam which consisted of 5 essay questions that covered few areas of SWEBOK.

For the course project at the professor's suggestion each of us chose his/her weakest SWEBOK topic, researched it and presented to the class.

This course was compared with similar courses at the University of Phoenix BSA385 Intro to Software Engineering and Rensselaer Polytechnic Institute ECSE-6770 Software Engineering and CISH-6050 Software Engineering Management. Pace University's course is tightly coupled with CSDA requirements, while other universities provide combination of theory and application.

SE-796 Software Research Seminar

This course covered software engineering topics depending on student's interests, weaknesses, and needs. Each student picked a topic perceived to be his or her weakest area and made a presentation to the class on that topic. The topic chosen was to be one of the topics covered in the IEEE CSDA exam. Since many of the students shared common weak areas, the rest of the class benefitted from each student's presentation. Student presentations were made by each individual student; students did not form teams to work on presentations. Because of the flexible and independent nature of this course, it was not feasible to compare it to comparable courses at other universities. The subject matter and depth of coverage was almost entirely up to the students.

5 Conclusion

Overall we think that this has been an excellent program. The partnership between BNY Mellon and Pace University in designing the curriculum assured us that the degree we earn would teach us skills that are valued in our workplace. The pace of the program, one class at a time and 3 classes per calendar year, allowed us to complete the program in three and a half years without excessively impacting our work and personal lives. It also allowed our Tuition to be fully covered under the BNY Mellon Tuition Assistance program. Taking these courses with other BNY Mellon employees and instructors who had signed a nondisclosure agreement allowed us to discuss real-life situations in class, enhancing our educational experience. We also met other people and learned about other things that are going on in technology at BNY Mellon through our interactions inside and outside of class.

6 References

[1] Detailed Course Information (IT 668) Available from https://banners.pace.edu/prod/bwckctlg.p_disp_course_detail?cat_term_in=201220&subj_code_in=IT&crse_num_in=668

[2] Detailed Course Information (CS 865) Available from https://banners.pace.edu/prod/bwckctlg.p_disp_course_detail?cat_term_in=201220&subj_code_in=CS&crse_num_in=865

[3] Detailed Course Information (SE 679) Available from https://banners.pace.edu/prod/bwckctlg.p_disp_course_detail?cat_term_in=201220&subj_code_in=SE&crse_num_in=679