

Strategies for Delivering Active Learning Tools in Software Verification & Validation Education

Dr. Sushil Acharya, Robert Morris University

Acharya joined Robert Morris University in Spring 2005 after serving 15 years in the Software Industry. His teaching involvement and research interest are in the area of Software Engineering education, Software Verification & Validation, Data Mining, Neural Networks, and Enterprise Resource Planning. He also has interest in Learning Objectives based Education Material Design and Development. Acharya is a co-author of "Discrete Mathematics Applications for Information Systems Professionals- 2nd Ed., Prentice Hall". He is a member of Nepal Engineering Association and is also a member of ASEE, and ACM. Acharya was the Principal Investigator of the 2007 HP grant for Higher Education at RMU. In 2013 Acharya received a National Science Foundation (NSF) Grant for developing course materials through an industry-academia partnership in the area of Software Verification and Validation. Acharya is also the Director of Research and Grants at RMU.

Dr. Priya Manohar, Robert Morris University

Dr. Priyadarshan (Priya) Manohar Dr. Priyadarshan Manohar is a Professor of Industrial Engineering and Co-Director Research and Outreach Center (ROC) at Robert Morris University, Pittsburgh, PA. He has a Ph. D. in Materials Engineering (1998) and Graduate Diploma in Computer Science (1999) from University of Wollongong, Australia and holds Bachelor of Engineering (Metallurgical Engineering) degree from Pune University, India (1985). He has worked as a post-doctoral fellow at Carnegie Mellon University, Pittsburgh (2001 – 2003) and BHP Institute for Steel Processing and Products, Australia (1998 – 2001). Dr. Manohar held the position of Chief Materials Scientist at Modern Industries, Pittsburgh (2003 – 2004) and Assistant Manager (Metallurgy Group), Engineering Research Center, Telco, India (1985 – 1993). He has published over 55 papers in peer-reviewed journals and conferences including a 2007 Best Paper Award by the Manufacturing Division of American Society for Engineering Education (ASEE), three review papers and three book chapters. He has participated in numerous national and international conferences. He is a member of ASM International, TMS, ACerS, AIST, ASEE, and a registered Chartered Professional Engineer. Dr. Manohar's research interests include mathematical and computer modeling of materials behavior, thermo-mechanical processing of steels and other metallic materials, microstructural characterization, and structure – property relationships. He has conducted a number of technical failure investigations, consulted on various materials-related problems, and acted as an expert witness in the Court of Law. Dr. Manohar is the past chair of the Manufacturing Division of ASEE and ASM Pittsburgh Chapter.

Prof. Peter Y Wu, Robert Morris University

Peter Y. Wu is professor of Computer and Information Systems at Robert Morris University. He earned Ph.D. in Computer System Engineering from Rensselaer Polytechnic Institute. He worked for IBM Research Division, first as a post-doc research fellow and subsequently a staff member at the T.J. Watson Research Center. He was the chief software engineer and a founding partner of UJB Solutions, LLC, a consulting company in production planning, for two years. He previously held faculty appointments at the Hong Kong Polytechnic University, and the University of Pittsburgh. His current research interests are in software engineering, internet computing, and geographic information systems.

Dr. Bruce R Maxim, University of Michigan, Dearborn

Bruce R. Maxim has worked as a software engineer, project manager, professor, author, and consultant for more than thirty years. His research interests include software engineering, human computer interaction, game design, social media, artificial intelligence, and computer science education. Dr. Maxim is associate professor of computer and information science at the University of Michigan—Dearborn. He established the GAME Lab in the College of Engineering and Computer Science. He has published a number of papers on computer algorithm animation, game development, and engineering education. He is coauthor

of a best-selling introductory computer science and software engineering texts. Dr. Maxim has supervised several hundred industry-based software development projects as part of his work at UM-Dearborn.

Strategies for Delivering Active Learning Tools in Software Verification & Validation Education

Imparting real world experiences is often a challenge due to both lack of effective active learning tools and effective delivery strategies. This pedagogical requirement is important because graduates are expected to develop software that meets rigorous quality standards in functional and application domains with little to no training. Lack of effective active tools has been addressed by the authors by designing, developing, and delivering, twenty (20) delivery hours of Case Studies, sixteen (16) delivery hours of Class Exercises, and six (6) delivery hours of Video Case Studies for use in courses that impart knowledge on SV&V topics viz. *requirements engineering, software reviews, configuration management, and software testing*. Four key skill areas sought after by employers, namely *communication skills, applied knowledge of methods, applied knowledge of tools, and research exposure* have been used to drive the development funded by a National Science Foundation grant and perfected through an industry-academia partnership. These tools have been successfully disseminated to over 25 universities with many CS, IS, SE programs incorporating the tools in their existing courses and others designing new courses based on these tools.

Since student learning retention depends on knowledge retention activities delivered in different settings it is important to identify and incorporate delivery strategies that meet course and/or module learning outcomes. In this paper we discuss strategies used to effectively incorporate and deliver the developed Active Learning tools by instructors at two universities. Traditional and flipped classroom delivery strategies are discussed as well as topics like pre-requisite knowledge preparation prior to class, course module presentation sequence, homework, team/individual work, collaborative discussions, and assessment tools are deliberated. In addition student feedback and assessment are also presented for each category of active learning tools and strategies that work are summarized.

1. Introduction

Imparting real world experiences is often a challenge due to both lack of effective active learning tools and delivery strategies. This pedagogical requirement is important because graduates are expected to develop software that meets rigorous quality standards in functional and application domains with little to no training. Institutions of higher learning across the nation are responding to political, economic, social and technological pressures to be more responsive to students' needs and more concerned about how well students are prepared to assume future societal roles ^[1]. The lack of effective active learning tools has been addressed by the authors at Robert Morris University by designing, developing, delivering, and disseminating twenty (20) delivery hours of Case Studies, sixteen (16) delivery hours of Class Exercises, and six (6) delivery hours of Video Case Studies for use in courses that impart knowledge on SV&V topics viz. *requirements engineering, software reviews, configuration management, and software testing*. Four key skill areas sought after by employers, namely *communication skills, applied knowledge of methods, applied knowledge of tools, and research exposure* have been used to drive the development funded by a National Science Foundation (NSF) grant and perfected through an industry-academia partnership. These tools have been successfully disseminated to over 25 universities with many CS, IS, SE programs incorporating the tools in their existing courses and others designing new courses based on these tools.

As student learning retention and eventual application of the knowledge gained depends on knowledge retention activities delivered in different settings it is important to identify and incorporate delivery strategies that meet course and/or module learning outcomes and enhances student learning. Researchers in both academia and industry have published literature on delivery strategies. Some of the more prominent strategies used in higher education are lecture, case method, discussion, active learning, cooperative learning, integrating technology and distance learning ^[1]. In this article our focus will be on both Teacher Centric and Student Centric methods with a focus on the effective and efficient delivery of the active learning tools that we have developed. The active learning tools are the results of an industry-academia partnership with an understanding that the knowledge and experience gained through these tools would be directly applicable in the workplace. Hence to effectively impart required knowledge appropriate and effective delivery strategies should be used.

In this paper we discuss strategies used to effectively incorporate and deliver the developed active learning tools by instructors at two universities. Traditional, flipped and hybrid classroom delivery strategies are discussed and topics like pre-requisite knowledge preparation prior to class and, course module presentation sequence, homework, team/individual work, collaborative discussions, and assessment tools are deliberated. In addition student feedback and assessment are also presented for each category of active learning tools and strategies that work are summarized.

2. Active Learning Tools

Active learning is “embodied in a learning environment where the teachers and students are actively engaged with the content through discussions, problem-solving, critical thinking, debate or a host of other activities that promote interaction among learners, instructors and the material” ^[2]. Prince ^[3] defines active learning as a classroom activity that requires students to do something

other than listen and take notes. Active learning is achievable by complementing lecture materials with case studies, class exercises, and case study videos.

2.1. Case Studies

Case studies can serve as useful tools to teach applications of science and engineering principles. Raju and Sankar ^[4] define case study education as providing students with a record of a technical and/or business issue that actually has been faced by managers, together with surrounding facts, opinions, and prejudices upon which management decisions have to depend. Using case studies as a semester-long tool to teach neuroanatomy, in which students were actively engaged in the presentation and discussion of case studies throughout the semester, resulted in more understandable and enjoyable learning experience for the students. In a study at Middlesex Community College ^[5], case studies were used in teaching General Biology - I where 88.2% of the students surveyed found the cases to be useful or better for learning the course content. 90.9% of the students surveyed thought the cases were useful or better in making the course more interesting. Case studies were applied in six courses to help students (1) understand complex and complicated issues and describe interrelated processes; (2) discuss policy- and decision-making ideologies that either are politically or socially charged; and (3) engage in informative and focused classroom discussion. The results indicated that use of the case study method as an active learning tool provides students with a variety of important skills necessary for success both in and out of the classroom. Specifically, active learning helps students develop problem-solving, critical-reasoning, and analytical skills, all of which are valuable tools that prepare students to make better decisions and become better students and, ultimately, better employees ^[6]. Raju and Sankar ^[4] undertook the research to develop a teaching methodology to bring real-world issues into engineering classrooms. The results of their research led to recommendations to funding agencies and educators on the need to develop interdisciplinary technical case studies so that the innovations happening in the field of engineering can be communicated to students in the classroom. The authors have disseminated twelve Case Studies. Examples of some of our Case Studies are Understanding User Requirements, A Software Test Plan, and Importance of Peer Reviews. A complete set of case studies has been published by Acharya and Manohar ^[7].

2.2. Class Exercises

Class exercises provide class time to explicitly raise questions that invite student participation. When well designed for the context and presented in the right setting, class exercises raise questions for the students to exercise their thinking. Depending on the focus of the questions, the students may be more motivated to investigate the subject matter, may gain a deeper understanding of course concepts, or may improve their skills through hands-on experience using the knowledge in problem solving and design derived from the exercises. There are many ways of using class exercises in the classroom setting. For a small class size, the teacher may simply use an exercise to engage students in discussion and hands-on practice. For larger classes, the students can be assigned to small groups using the class exercise as an instrument leading to group projects. Woods and Howard ^[8] effectively used class exercises for Information Technology students to study ethical issues. Day and Foley ^[9] used class time exclusively for exercises, having their students prepare beforehand for class with materials provided online. Bishop and Verleger ^[10] presented a comprehensive survey of the research that reviewed different ways of using class exercises in the classroom, often referred to as the "flipped

classroom." Frydenberg^[11] primarily used hands-on exercises to foster student understanding in data analytics. Well designed, class exercises become very effective learning tools and can be versatile in various classroom settings. The authors have disseminated sixteen exercises^[12]. Examples of some of our Class Exercises are Requirements Ambiguity, Defect Lifecycle, and IEEE Standards.

2.3. Case Study Videos

One commonly used technique to enhance the classroom learning experience is the use of video. Videos are viewed as an effective method of presenting standard material while addressing students of different learning styles. A video actively engages visual learners with its images and motions, while auditory learners can listen carefully to the narration to gain an understanding of the topic. Videos are an essential part of the flipped classroom model (discussed later), in which the preponderance of lecture material is presented before class^[13]. The class time is then spent on discussion and teamwork, reinforcing the material from the previous session. However, videos can also be used in a traditional classroom, and their use can be highly effective. There is extensive experience in using audio visual materials in the classroom, ranging from the usage of filmstrips during World War II to train soldiers^[14] to modern digital video. Watching videos can reinforce reading and lecture material, help to develop common knowledge, enhance the quality of discussion and overall student comprehension, accommodate students of different learning styles, increase student motivation, and increase teacher effectiveness^[15]. Videos can aid in showcasing highly complex concepts and ideas in a short period of time, provoking meaningful discussion and analysis. The video scripts were first drafted by the industry partners and subsequently confirmed by the testimonies shared in focus group discussions. The process has been described in details in a previous publication^[16]. Four videos, ranging from 15 to 24 minutes, have been disseminated. Examples of some of our Video Case Study titles are Formal Inspection Scenes and Requirement Analysis Scenes.

3. Delivery Strategies

Mishra et al.^[17] report that prior knowledge about software quality concepts and good programming skills can help students to achieve success in a software verification & validation course. Furthermore they also suggest that team work can be chosen as a strategy, since it facilitates students' understanding and motivates them to study. With the active learning tools designed to impart practical knowledge into theoretical understanding, we encourage a flipped classroom model^[18] in which we can maximize class time for active learning tools so as to engage the students for further digestion of the knowledge in the context of industry practices. Students are expected to be prepared outside of the classroom beforehand, with assigned textbook readings or reviewing of online materials. For effective delivery it is also recommended that students work in small teams. Overall, the flipped classroom model has proven highly effective at increasing student engagement and enhancing the preparation of students for class sessions^[9]. The flipped classroom also has been shown to allow the instructor to cover more material and results in higher student performance^[19].

The active learning tools engage the students in different ways to study SV&V in practice. The Case Studies are explicit in this approach: each Case Study makes the point to consider issues in realistic practices. Instructors can present the Case Study while guiding students into further

study and discussion of the practical issues in SV&V. The Class Exercises are designed for interaction in the classroom during group discussions. The instructor brings the question(s) and serves as a moderator to guide the discussion session. The instructor may also use the Class Exercise to lead students into subsequent group mini projects or individual mini projects. The “Instructor Notes” component of the Class Exercise discusses these possibilities. Students are likely to find the Case Study Videos by their nature as multimedia, as highly engaging. These videos share real-life perspectives of actions and their consequences. The videos are by design in sequences of scenes. For instructional purposes, it is highly beneficial to “pause” the video at appropriate moments so as to engage the students in discussion. Suggested discussion questions accompany the videos.

The active learning tools are modularized into flexible modules of 25 delivery minutes each. Instructors may consider their various needs such as curriculum design, class time, and class size to adapt the active learning tools to the situations in their institutions. Furthermore, though we recommend it, instructors do not need to practice the flipped classroom model or may do so partially.

To summarize, the course delivery plan encourages the following:

- Using the flipped classroom model (if applicable)
- Have students work in small teams (2-3 students per team)
- Delivering tools in one or multiple 25-minute sessions
- Using pre-test and post-test instruments to tailor course delivery
- Evaluating student learning of the module immediately after delivery

4. Active Learning Tool Description and Dissemination

To ensure instructors have the resources they need to start a new course or strengthen existing courses an instructor kit has been created. To ensure instructors have the adequate resources they need to teach a new module each module at a minimum comprises of the following components as separate MS Office file:

- a) ***Active Learning Tool Description:*** This is a MS Word file that provides a complete description of the active learning tool. This file consists of the following sections
- b) ***Instructional Slide:*** This is a MS Power Point file that provides instructional support to the instructor. This file is available for most of the tools. Instructors are advised to use this as it is or customize it to suit their needs.
- c) ***Student Handout:*** This is a MS Word file that can be customized and handed out to students during class. Students either return the completed version of this file or develop new documents that are then submitted for assessment.
- d) ***Assessment Instrument:*** This is a MS Word file that is used for assessing student learning (Figure 1). The survey has 10 questions and the host institutes Institutional Review Board (IRB) approval number. The questions were prepared by the project’s external evaluator and an expert in assessments. Institutions are requested to customize this file to include their institution and course names. The survey is to be assessed as soon as the delivery of the active learning tool is complete.

In addition the kit consists of sample syllabus, pre-tests/post-tests and answer key, and samples of midterm/ final exams.

RMU-IRB Approval #130102

Student Name (Optional):

<University Name>
 <Course Level>
 <Course Number> – <Course Name>
 Exercise Module: Testing Tools

Exercise Module: ST03

	Yes	No
I understood the purpose of the activity.	<input type="checkbox"/>	<input type="checkbox"/>
I could complete the activity with the directions that were provided.	<input type="checkbox"/>	<input type="checkbox"/>
I was uncertain of how to carry out the steps of the activity.	<input type="checkbox"/>	<input type="checkbox"/>
The activity used a real-world application.	<input type="checkbox"/>	<input type="checkbox"/>
I could imagine carrying out this activity as part of my job.	<input type="checkbox"/>	<input type="checkbox"/>
I communicated verbally in a <u>small group</u> while completing this activity.	<input type="checkbox"/>	<input type="checkbox"/>
I communicated verbally in a <u>large group</u> while completing this activity.	<input type="checkbox"/>	<input type="checkbox"/>
I / we provided written communication as part of this activity.	<input type="checkbox"/>	<input type="checkbox"/>
I / we made a formal presentation as part of this activity.	<input type="checkbox"/>	<input type="checkbox"/>
I thought critically about content while completing this assignment.	<input type="checkbox"/>	<input type="checkbox"/>
Provided your overall thoughts about this activity:		

Figure 1: Assessment Instrument

The active learning tools are available through the project website www.rmu.edu/nsfvv (depicted in Figure 2) and ENSEMBLE, a Computing Portal connecting Computing Educators, accessible through www.computingportal.org (depicted in Figure 3). The tools and supporting documents are organized based upon SV&V topics. Folders are provided for activities related to Configuration Management, Requirements Management, Software Reviews, and Software Testing. Underneath each of these folders are folders for the active learning tools: Case Studies, Class Exercises, Case Study Videos, and Topical Assessments. For greater availability, the videos have been uploaded to YouTube.

Award # 1245036



Collaborative Education:
Building a Skilled V&V Community
 Engineering Department, Robert Morris University
 Moon Township, PA , USA
 PI: [Dr. Sushil Acharya](#)
 Co-PIs: [Dr. Priya Manohar](#) & [Dr. Peter Wu](#)



Search this site

HOME
ABSTRACT
OBJECTIVES
TIMELINE
RMU TEAM
ACADEMIC PARTNERS
INDUSTRY PARTNERS
CONFERENCES/JOURNALS

SELECT A V&V TOPIC:

[REQUIREMENTS MANAGEMENT](#)

[SOFTWARE TESTING](#)

[SOFTWARE REVIEWS](#)

[CONFIGURATION MANAGEMENT](#)

[ADDITIONAL TOPICS](#)



An Academia - Industry Partnership

This material is based upon work supported
 by the National Science Foundation under Grant Number 1245036

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

ROBERT MORRIS UNIVERSITY

6001 University Blvd. Moon Township, PA 15108 800-762-0097

Disclosures | Maps & Directions | Employment at RMU | Contact Us | Login

© 1995 – 2011 Robert Morris University. All rights reserved.

Figure 2: Active learning tools in project website

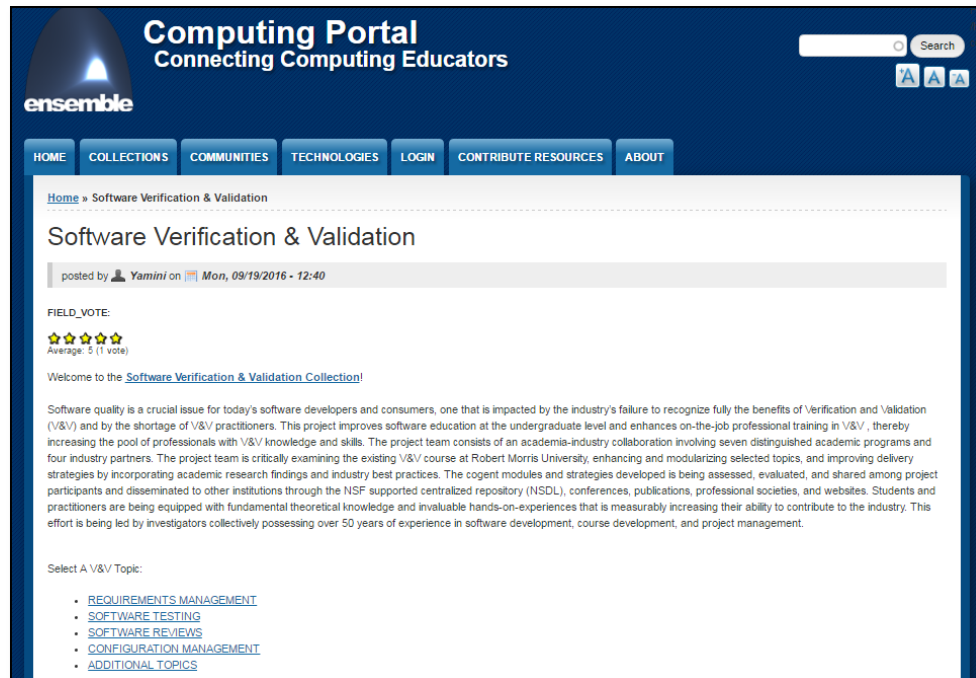


Figure 3: Active learning tools in ENSEMBLE

5. Institutional Profiles

As this article focuses on two distinctly different institutions and their different approaches to teaching software verification and validation, it is important to understand the differences in the institutions' profiles' and activities. This section provides background information on the two programs' profiles in this article.

a. Robert Morris University (RMU)

Robert Morris University offers an accredited Bachelors of Science degree in engineering (Software Engineering concentration), and has been accredited since 2002. The emphasis is on small class sizes (10:1 student to faculty ratio) and hands on experiences through class assignments, course projects, internships (150 hours mandatory), and an interdisciplinary capstone project (3 credits). RMU also offers B.S. in Engineering degrees in Mechanical, Industrial, and Biomedical Engineering concentrations and a B.S. in Manufacturing Engineering (the only one of its kind in Western Pennsylvania). A Software Engineering Lab is a part of the engineering departments' Learning Factory (lab space) and is used for software engineering class sizes of 8-10 students. As software engineering is offered as a concentration some of the required courses and a few of the elective courses are offered through RMU's Computer Information Systems Department (for example courses on programming languages, data structures, databases, etc.). All software engineering majors take the same set of courses to complete 117 credits out of the required 126 credits. For the remaining 9 credits which are software engineering elective courses students are free to take courses in the areas of their interests like network security, databases, etc. All software engineering courses offered by the engineering department are delivered in a 2+2 hours or 1+3 hours format for a total of 4

meeting hours per week. Even though the courses are 3 credits the extra meeting hour is used for additional hands on experiences.

b. University of Michigan - Dearborn (UMD)

The University of Michigan - Dearborn (UMD) is one of three campuses of the University of Michigan and has an enrollment of approximately 9000 students. The Department of Computer and Information Science (CIS) at the University of Michigan - Dearborn is one of four departments housed in the College of Engineering and Computer Science. CIS offers accredited Bachelors of Science degrees in both Software Engineering (SE, accredited 2002) and Computer and Information Science (CIS, accredited 1997). CIS also offers undergraduate degrees in Digital Forensics (DF) and in Data Science (DS). We plan to seek accreditation of these two degrees after we have our first graduates. CECS also has several masters and doctoral programs in which CIS is a participant. The primary mission of the CIS department is to provide high-quality computing education to students in the Detroit metropolitan area. Our graduates are highly sought out by potential employers and work at major companies throughout the United States. All CIS undergraduates are required to take 120 credit hours of course which includes 58 credit hours in the CIS department. Most CIS courses are offered as lecture classes with 4 contact hours each week. There are several classes which also require attendance in closed laboratory sessions. The emphasis in CECS is on small class sizes (30:1 student to faculty ratio) and offering hands on experiences through class assignments, course projects, internships, and a two semester (8 month) capstone design project course (4 credits total). The CIS capstone project experience involves students working as part of a four person team under the mentorship of a senior CIS professor to develop a software solution to a problem posed by an industrial partner.

6. Course Delivery

The following narrative discusses where the active learning tools are incorporated in the curriculum of the two institutions being discussed.

a. RMU

ENGR3400: Software Verification and Validation is one of the three original software engineering courses incorporated in the software engineering curriculum on the behest of the Engineering Program Advisory Committee (comprised of local industry partners). Prior to 2005 this course focused highly on testing. In 2005 the course was overhauled to incorporate *requirements engineering*, *software reviews*, *configuration management*, and *software testing* to provide student's knowledge relevant to the software development industry. To ensure adequate contact hours to cover the contents contact hours for this 3 credit course was increased to 4 hours. After the active learning tools were developed in 2014 the course is being taught using the recommended flipped classroom model with majority of the 4 contact hours being used for engaged/experiential learning. 40% of the total student grade is allocated to class assignments. This has helped students to prepare and be punctual in class. Student performance in this course is measured in terms of ABET outcomes. A class assessment performed in Spring 2013 when case studies were not available showed a weakness in learning outcome 'e' (an ability to identify, formulate, and solve engineering problems) where less than 60% of the students scored better than 80% in the assessment tasks. When the case studies were delivered in the

Spring 2015 term the student performance in outcome 'e' was excellent range ($\geq 90\%$). This presents clear evidence that the case study based teaching method is more effective in delivering an ability to identify, formulate, and solve engineering problems to the students. The developed tools have been taught in spring 2014, spring 2015, and spring 2016 semesters. Experience from previous delivery is used in course planning for the succeeding delivery. Likewise findings from the assessments are used to revise the developed active learning tools.

b. University of Michigan - Dearborn (UMD)

CIS 375: Software Engineering 1 is the first software engineering course taken by all SE, CIS, and DF majors. This course was created in 1997 as a general software engineering lecture course and is currently offered as 4 credit hour class. The topics covered include software process models, requirements engineering, requirements modeling, project management, configuration management, software quality, software architecture and design, user experience design, design patterns, and testing. Students complete a team software development project and write several milestone documents as part of the project deliverables. During 2016 active learning tools, developed at RMU, were examined, and adapted for use in CIS 375. Several additional activities were developed at UMD to cover topics outside the scope of ENGR 3400 as taught at RMU. In the Fall 2016 semester CIS 375 was taught using a flipped classroom model with majority of the 4 contact hours being used for engaged/experiential learning. 5% of the total student grade is allocated to class participation and 75% was allocated to term project deliverables. The remaining 20% of the student grade came from the two in class examinations. Experience from this delivery is being used to revising the next offerings of this course and the active learning materials.

7. Delivery Strategies at RMU and UMD

In the following sections we discuss how applicable active learning tools are delivered in these institutions.

7.1. Strategy for Delivering – Lectures

a. RMU

Blackboard Learning Management System is used at RMU as the main delivery tool for on-line programs as well as a course repository for on-ground program. Fourteen weeks of course contents for ENGR3400 are uploaded to a Blackboard shell and then shared with the class prior to the start of the semester. Figure 4 below depicts the structure of Week 7 course contents in Blackboard.

- i. **Homework:** As the course is being delivered in a flipped classroom model, prior to coming to class students are required to read the course slides relevant for that day or week. Each course slides begins with learning outcomes and ends with possible discussion questions. Should students have any questions prior to class they are requested to email the instructor or to meet the instructor during office hours to discuss the question(s).
- ii. **Classwork:** In class the instructor spends the initial 5-10 minutes discussing a subset of the discussion questions. Randomly selected students are asked

questions and the instructor supplements the responses by providing additional thoughts. At times the instructor also carries out mini lectures (10 mins) to reinforce key concepts. At the end of an active learning tool exercise students complete a survey which takes about two minutes to complete.

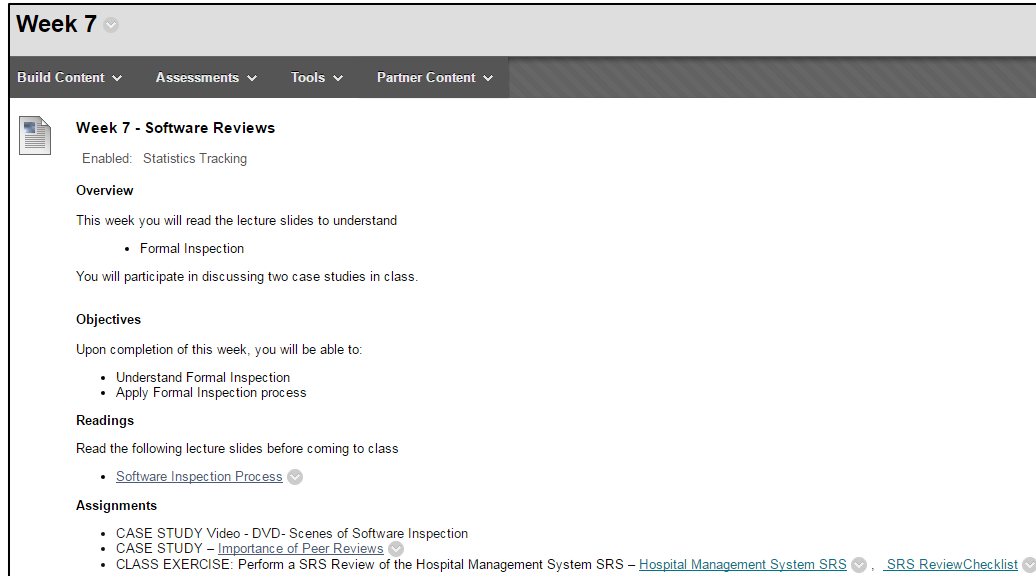


Figure 4: Course contents in Blackboard

b. UMD

- i. **Homework:** Prior to coming to class students are required to read the sections of the course text book (Pressman and Maxim, 2015) assigned for that day. The lecture slides and class handouts were available on UMD's Canvas course management system. Should students have any questions prior to class they are requested to email the instructor or to meet the instructor or teaching assistants during their office hours.
- ii. **Classwork:** In class the instructor spends the initial 30 minutes introducing the day's topics and activities. Most class periods students work in small groups to complete the day's active learning tasks. At the end of an active learning exercise students complete a survey that takes about two minutes to complete.

7.2. Strategy for Delivering – Case Studies

a. RMU

In 2016 twelve Case Studies were successfully delivered to a class of 22 students in a room that was equipped with discussion tables and computers.

- i. **Homework:** Some of the case study descriptions are given to the students as home reading assignments. With this approach the students come to class better prepared as they have a better understanding of the case study exercise that would be conducted that day. When time became an issue the concluding part of a case study exercise is given as home assignment. For example refining the Software pressmans Specifications (SRS) document, and developing a presentation for class.

- ii. **Classwork:** The case study starts with students reading the case study description. The instructor also explains the case study to the class. This usually takes 5 minutes. Next the student teams are formed. Randomness is used to form this team so that the students don't always work with the same person. This gives the students new perspectives and also forces them to leave their comfort zone. The case study results in a power point presentation or a discussion. Students are requested to use the room computers or their own should they need to research on a specific area. For example to make a list of open source configuration management tools they will search reliable websites. Student teams make the presentation with every team member participating in the presentation. Students and instructor both ask questions. For discussion question, questions are projected and responses from different teams are requested.
- iii. **Student Performance and Feedback:** Teamwork, discussion participation, presentation, and work products were used to measure student learning. All students liked the hands-on experience, the teamwork and the team presentations. They felt they remained alert and were not distracted. Students felt they communicated well in their team settings and thought critically about content while completing the assignments. Students also felt the presentations of their work to the class gave them a sense of achievement and helped them polish their communication skills. Some student comments on the case studies were:
 - *It gave real world examples of the legal aspects of the software which gave me a better understanding of the field.*
 - *Was a good introduction for real life application*
 - *Very interesting an eye opening to how far people will take the legalities of their agreements.*
 - *The activity allowed my group member and I to share thoughts. It was to the point and educational.*
 - *The scenario was setup in a way that was easy to understand.*
 - *Very creative and effective. It helps to learn more than the old way.*

b. UMD

In 2016 seven Case Studies were successfully delivered to a class of 36 students in a room that was equipped with discussion tables, multiple monitors, and portable white boards. Case studies were completed within the two hour class period.

- i. **Homework:** Prior to coming to class students read the sections of the course text book ^[20] assigned for that day. The lecture slides and class handouts were available on UMD's Canvas course management system.
- ii. **Classwork:** The instructor spent 30 minutes introducing the day's topics in class. Students worked in small groups to complete the day's active learning tasks. Student groups made informal presentations to summarize their group work. The instructor lead the class in a debrief.
- iii. **Student Performance and Feedback:** The student performance on the case studies was best measured in their term project work, especially the software requirements document and the test plan document. The instructor felt their

work was superior to what had been observed in the past from students in this class, but no statistical tests were performed. The student comments indicated that they enjoyed the case study activities and felt that they prepared them for work required to create the term project deliverables. They also felt that sharing ideas and insights with other students during class discussions helped them learn. They enjoyed being able to apply the material covered in the textbook to solve typical work place problems.

7.3. Strategy for Delivering – Class Exercises

a. RMU

In 2016 fourteen Class Exercises were successfully delivered. Though the exercises are usually 25 minutes long the delivery duration depends on the class size. For a class of 22 students 8 exercises were completed in 4 class periods and the remaining 6 exercises took the entire period i.e. 50 mins.

- i. **Homework:** For class exercises, there is no homework. Students are made aware that the exercises depend on the lecture materials so they are encouraged/expected to read prior to coming to class.
- ii. **Classwork:** All class exercises are completed in class. The instructor spends the first 3-5 minutes introducing the exercise and then distributes the student handouts. Students are usually given 10-15 minutes to complete the exercise alone. However in certain exercises students are given 20-25 minutes. The questions are then discussed in class. In most of the exercises there is no teamwork involved. However in some exercise student work in teams. If the student's response is not correct they are encouraged to make corrections keeping intact their earlier response. When class finishes students are asked to submit the assignments for grading. The answer sheets are returned to the students in the next class.
- iv. **Student Performance and Feedback:** Teamwork and discussion participation were used to measure student learning. All students liked the hands-on experience and the work carried out small teams of 2 or 3. They felt they understood what was expected and made use of lecture slides and past discussions to answer the questions. Given the opportunity to discuss their findings many felt it was good to share their thoughts and hear the thoughts of others. Most important it was understood that answers to some questions could vary as software development problems are not like calculus problems i.e. everyone arrives at the same answer. Some student comments on the class exercises were:
 - *Each group presented their thought on the activity. The activity also did a great job in teaching real-world situations.*
 - *Interesting and provided needed perspectives*
 - *It made me feel like I am in the industry. Also it opens my mind to imagine my future career.*
 - *It was a good intro into real life application.*
 - *This activity showed how little hiccups can turn into big problems.*
 - *This was good instruction into the relationship between developers and customers after deployment.*

- *Great activity that made me look at different situations from real life.*
- *I thought it was a good activity to get me to start thinking critically about the subject.*

b. UMD

In 2016 thirteen Class Exercises were delivered. Most of the exercises were 25 minutes long so that more than one may be delivered in the same class period. Some of the exercises were designed to take 50 minutes to complete.

- Homework:** Prior to coming to class students read the sections of the course text book (Pressman and Maxim, 2015) assigned for that day. The lecture slides and class handouts were available on UMD's Canvas course management system.
- Classwork:** The instructor spent 30 minutes introducing the day's topics in class. Handouts were distributed and students worked in small groups to complete the day's active learning tasks. Student groups made informal presentations to summarize their group work. The instructor lead the class in a debrief.
- Student Performance and Feedback:** Student performance on class exercises was assessed using two 90 minute written examinations. The class average for the first exam was 83.9% and for the second exam was 83.6%. The students liked the use of multiple diagrams to represent requirements. They also enjoyed writing meaningful questions during requirements engineering activities to help resolve the ambiguities inherent in working with customers. They felt these activities were more engaging than just listening to a lecture accompanied with slides. The students liked the redundancy that was built in the activities that often had them look at different facets of the similar problems. They felt the group work and subsequent presentation summaries help them to improve their communications skills. They enjoyed the group work and loved using the active learning classroom. They liked being able to critique testing artifacts created by others. Occasionally they would have liked more time to complete an activity.

7.4. Strategy for Delivering – Case Study Videos

a. RMU

In 2016 three case study videos were successfully delivered in class.

- Homework:** For Case Study Videos there is no homework.
- Classwork:** Case Study Video assignments are worked on during class time. Each video has multiple scenes and for each scene a set of possible discussion questions are available to the instructor. For this exercise no teams are formed, students work alone but the whole class participates in the discussion. Students are given a description document that includes a set of discussion questions for the first scene of the video being discussed. Providing the questions helps the students view the first scene with the questions in mind. For the subsequent scenes questions are shared after the scenes are viewed. After each scene the class discusses the happenings of the scene. Instructor questions are used to start the discussion. Instructor randomly selects a student and asks the question. Other students are asked to weigh in on the answer. The instructor supplements

the answer with his thoughts. After the discussion concludes the next scene is played.

- iii. **Student Performance and Feedback:** Discussions after each scene were used to measure and understand student learning and perspective. The students felt that scenes and the dramatization of the software engineering process helped them relate to industry best practices. The questions asked, the responses from their classmates, and the thoughts of the professor helped them understand why certain things would happen at work. With the understanding of the “dos” and “don’ts” as depicted in the video students felt confident on being able to execute in the real world should situation arise. Some student comments on videos shown in class were:

- *Inightful exercise for understanding the process.*
- *It did a good job at demonstrating requirement analysis while dealing with a client.*
- *The video was professional a real-life situation. Maybe include more group thinking.*
- *Gives you a good picture of the real-world. Work place is not always friendly. However you must be a professional.*
- *Was good to explain how V&V would be like in real world.*
- *Very helpful in understanding how to deal with customers.*
- *It was a descent representation of a how a real client interaction takes place.*
- *I thought it was an effective way to facilitate a discussion*

b. UMD

In 2016 five case study videos were delivered in class.

- i. **Homework:** For Case Study Videos, there is no homework.
- ii. **Classwork:** Case Study Video assignments were worked on during class time. Each video has multiple scenes and each scene has a set of discussion questions. For this exercise, no teams were formed, students viewed the video as a class and participated in the discussion of each scene.
- iii. **Student Performance and Feedback:** We used the video case studies as trigger films to provide a context for class discussions. The students appreciated the irony and humor presented in the videos. In many ways, the videos showed why it is important to do things right the first time and the students thought that was good. By the end of the semester, they made many comments about understanding the importance of quality as guiding principle that needs to be pervasive thought a software development organization.

8. Sample Assessment Results

Table 1 captures student feedback at RMU after a successful delivery of a case study module titled “Understanding User Requirements”. The class size was 29 however some students did not turn in the assessment and few students were absent on the day this exercise was delivered. 40% of the students were also international with English not being their first language. They had some issues understanding the survey questions. The assessment results

show that most of the students who were in the class when this case study exercise was conducted understood the exercise and were able to complete it.

Table 1: Student Feedback at RMU for a Case Study Module

Question	Yes	No
My team and I understood the purpose of the activity.	23	
My team and I could complete the activity with the directions that were provided.	23	1
At least one member of my team was uncertain of how to carry out the steps of the activity.	4	19
The activity used a real-world application.	23	
I could imagine carrying out this activity as part of my job.	22	1
I communicated verbally in a <u>small group</u> while completing this activity.	22	1
I communicated verbally in a <u>large group</u> while completing this activity.	3	20
I / we provided written communication as part of this activity.	22	1
I / we made a formal presentation as part of this activity.	1	22
I thought critically about content while completing this assignment.	23	
Provided your overall thoughts about this activity: <ul style="list-style-type: none"> - This activity showed us how there is a lot of short cuts we take when speaking. We need to be able to clearly describe what we want and need. - Group discussions are helpful only if you are in the right group. - After this exercise, I have a good idea of writing a requirement. - This activity allowed me and my teammate to share thoughts. 		

Table 2 captures student feedback at UMD after a successful delivery of an exercise module titled “Test Cases for a given requirement”. The class size was 35 however only 23 returned the surveys. A few students were also absent. The assessment results show that most of the students who were in class when this exercise was conducted understood the exercise and were able to complete it. They also had to work in smaller groups and then participate in the class discussion. Students who provided their thoughts were supportive of the exercise.

Table 2: Student Feedback at UMD for an Exercise Module

Question	Yes	No
My team and I understood the purpose of the activity.	21	
My team and I could complete the activity with the directions that were provided.	21	
At least one member of my team was uncertain of how to carry out the steps of the activity.	2	19
The activity used a real-world application.	20	1
I could imagine carrying out this activity as part of my job.	19	2
I communicated verbally in a <u>small group</u> while completing this activity.	19	2
I communicated verbally in a <u>large group</u> while completing this activity.	13	8
I / we provided written communication as part of this activity.	6	15
I / we made a formal presentation as part of this activity.	1	20
I thought critically about content while completing this assignment.	20	1
Provided your overall thoughts about this activity: <ul style="list-style-type: none"> - Good idea for those who have less experience - Good exercise - It was good - Helped clear up the test case issue 		

9. Conclusion and Recommendations

As student learning retention depends on knowledge retention activities delivered in different settings it is important to identify and incorporate delivery strategies that meet course and/or module learning outcomes. In this paper we discussed the flipped classroom strategy that was used to effectively incorporate and deliver the active learning tools developed at RMU and customized to suit their needs at UMD. Teamwork, discussion participation, presentation, and work products were used to measure student learning. Instructors felt student work was superior to what had been observed in the past from students in their classes. Students surveyed liked the hands-on experience, the teamwork, and the team presentations. They felt they remained alert and were not distracted. Students also felt they communicated well in their team settings and thought critically about content while completing the assignments. Student comments indicated that they enjoyed the activities and felt that these activities would help them for future work in these areas.

10. Acknowledgement

The authors acknowledge the support of NSF through a grant entitled “Collaborative Education: Building a Skilled V&V Community”, NSF-TUES Award # 1245036.

References

- [1]. Teaching Strategies, [Web Page], Retrieved January 15, 2017 from URL <http://www.gmu.edu/resources/facstaff/part-time/strategy.html>
- [2]. Promoting Active Learning, [Web Page], Retrieved January 21, 2017 from URL <https://utah.instructure.com/courses/148446/pages/active-learning>
- [3]. Prince, M., “Does Active Learning Work? A Review of the Research,” *Journal of Engineering Education*, Vol. 93, 2004, pp. 223-231.
- [4]. Raju, P. K. and Sanker, C. S. (1999) Teaching Real-World Issues through Case Studies, *Journal of Engineering Education*. Vol. 88 No 4 pp501-508
- [5]. [Web Page], Retrieved December 2, 2016 from URL <https://www.middlesex.mass.edu/sotl/downloads/klein.pdf>
- [6]. Kunselman, J.C. and Johnson, K.A., Using the Case Method to Facilitate learning, *College Teaching*, Vol. 52. No. 3 (Summer 2004).
- [7]. Acharya, S., & Manohar, P. A. (2016). *Case Studies in Software Verification & Validation*. Alexander Street Press
- [8]. Woods, D., and Howard, E (2014) An Active Learning Activity for an IT Ethics Course. *Information Systems Education Journal*, 12(1) pp.73-77. <http://isedj.org/2014-12/> ISSN: 1545-679X.
- [9]. Day, J.A. and Foley, J.D. (2006) Evaluating a web lecture intervention in a human-computer interaction course. *IEEE Transactions on Education*, 49(4):420–431, 2006.
- [10]. Bishop, J.L. and Verleger, M.A. (2013) The Flipped Classroom: A Survey of the Research. ASEE 120th Annual Conference and Exposition, Atlanta, GA.
- [11]. Frydenberg, M. (2013) Flipping Excel, *Information Systems Education Journal*, 11(1) pp.63-73. <http://isedj.org/2013-11/> ISSN: 1545-679X.
- [12]. Wu, P., Manohar, P. A., & Acharya, S. (2016). The Design and Evaluation of Class Exercises for Active Learning in Software Verification and Validation. *Information Systems and Education Journal*, 14(4), 4-12.

- [13]. Bergmann, J. and Aaron S. (2012) *Flip Your Classroom: Reach Every Student in Every Class Every Day*, Eugene: International Society for Technology in Education, 2012. Print.
- [14]. Hovland, C.I., Lumsdaine, A.A. & Sheffield, F.D. (1949). Experiments on mass communication. Princeton, NJ: Princeton University Press.
- [15]. Corporation for Public Broadcasting. (2004). Television goes to school: The impact of video on student learning in formal education
- [16]. Acharya, S., Manohar, P. A., & Wu, P. (2016). *Using Case Study Videos to Effectively Teach Software Development Best Practices* (pp. 230-235). The 20th World Multi-Conference on Systemics, Cybernetics, and Informatics (WMSCI) Conference, Orlando, FL, Organized by International Institute of Informatics and Systemics (IIS).
- [17]. Mishra, D., Hacaloglu, T., & Mishra, A. (2014). Teaching Software Verification and Validation Course: A Case Study. *International Journal of Engineering Education*, 30, 1476-1485.
- [18]. Bonwell, C. C., & Eison, J. A. (1991). *Active Learning; Creating Excitement in the Classroom*. ASHE-ERIC Higher Education Report No. 1., Washington, D.C.: School of Education and Human Development, The George Washington University. Retrieved from <http://files.eric.ed.gov/fulltext/ED336049.pdf>
- [19]. Mason, G.S.; Shuman, T.R.; Cook, K.E. (2013), "Comparing the Effectiveness of an Inverted Classroom to a Traditional Classroom in an Upper-Division Engineering Course," *Education, IEEE Transactions on* , vol.56, no.4, pp.430,435, Nov. 2013
- [20]. Pressman, R. S. & Maxim, B. R. (2014) *Software Engineering: A Practitioner's Approach*, 8th Ed., McGraw-Hill.