Value Creation in Engineering Education through Goal Directed Course Design

JosAnn Duane[a], Ye Liu[a], and Laurie Maynell[b],
[a] Engineering Education Innovation Center
[b] University Center for the Advancement of Teaching
The Ohio State University, Columbus, Ohio 43210
Email: duane.1@osu.edu

Introduction and Objective:

This poster paper[1] presents the process of course design and summarizes the work of a group of poster papers detailing six aspects of course design. The course under development is a two credit hour, semester course on engineering problem solving using software tools primarily MATLAB[2]. Students enter the course with varies levels of expertise in using MATLAB. However, all have has some experience with MATLAB.

Our objective is development of this course through goal directed course design. We envisioned a course teaching the fundamental *thought processes* underlying problem solving, the software *tools* required for implementation, the creative thinking underlying *design*, the supporting professional *conduct* and teamwork, and basic engineering *communication* skills. The course aims to achieve the following ABET student outcomes.

ABET Student Outcomes

1. THINK: Competency in an ability to identify, and solve (*a specific subset of*) engineering problems; Familiar with an ability to apply knowledge of mathematics, science, and engineering; and, Exposure to the ability to design and conduct experiments, as well as to analyze and interpret data.

2. USE TOOLS: Competency in an ability to use the techniques, skills, and modern engineering tools (*specifically MATLAB*) necessary for engineering practice.

3. DESIGN: Competency in an ability to design a (*MATLAB software*) system, component, or process to meet desired needs.

4. CONDUCT: Familiarity with an ability to function on multi-disciplinary teams; and familiarity with an understanding of professional and ethical responsibility.

Method:

Both the iterative, incremental design method[3] and the backwards design method[4] were used to achieve our objective. Starting with an initial design, the iterative, incremental design method repeats the following cycle until the desired level of performance is achieved: Observe, test, and evaluate the design; define problems; develop solutions; and, implement the solutions. The iterative design emphasizes prototyping elements early in the design phase.

The following section describes how a set of goals and course learning objectives were derived from the given set of desired ABET student outcomes as listed below.

Course Goals and Learning Objectives

1. THINK:
Goal - *Fluency in engineering thought processes: critical, creative, practical thinking.*
Learning Objective - *Demonstrate ability in critical, creative and practical thinking through algorithm design, MATLAB software design and evaluation.* **For example**: Understanding computational science concepts, including simulation, optimization, and data analysis; Use of computational approaches in solving complex problems in science and engineering, and procedural composition.

2. USE TOOLS:
Goal - *Expertise in using engineering tools.*
Learning Objective - *Utilize MATLAB software tools to solve engineering problems.* **For example**: Writing simple MATLAB programs performing numerical calculations; Using basic constructs provided by high-level programming languages; Using basic data structures such as numeric and character arrays; Using interactive programming to develop, design and analyze software tools.

3. DESIGN:
Goal - *Ability to apply the principles of engineering design*.
Learning Objective - *Demonstrate the ability to create and design within the constraints of time, cost, quality, safety, and environmental impact.* **For example**: Employing methods such as "brainstorming" to generate possibilities and methods such as "pair and share" to evaluate possibilities; Utilizing tools such as budgets and schedule charts to implement designs.

4. CONDUCT:
Goal - *Professional conduct, values, and ethics of an engineer.*
Learning Objective - *Work individually, in pairs, and on teams to solve engineering design and analysis problems professionally and ethically.* **For example**: Understanding concepts of intellectual property; Understanding responsibility and accountability.

5. COMMUNICATE:
Goal - *Effective engineering communications.*
Learning Objective - *Demonstrate skill in technical communication related to engineering and software development.* **For example**: Communicating problem solving results using standardized methods of software documentation, flowcharts, and computer generated charts, graphs, and animations to present numerical results.

When using the method of backward design, course goals, statements of what students should know and care about upon successful completion of the course, are articulated first. We derived the course goals from a given subset of ABET student outcomes. Next, we developed observable and measurable learning objectives, which were aligned to the course goals.

In the third step of backwards design, an assignment-centered course outline is generated, aligned to learning objectives and with theories of student learning in mind. Lastly, teaching methods and content are added. Course goals and learning objectives determine all decisions regarding assignments, teaching methods and content.

A spreadsheet tool was created to implement backwards design using the iterative design method. Electronic, movable, color-coded icons were used to represent goals, learning objectives, feedback and evaluation methods, teaching methods, and course content. A portion of the tool, shown to the right of the poster paper[1] illustrates the progression of the course. The feedback and evaluation methods are used to measure students' achievement of learning objectives. As shown on the poster paper, these icons can be easily rearranged as the course proceeds through the stages of iterative, incremental design. The icons are color coded in alignment with the learning objectives of the course as follows: THINK (coral), USE TOOLS (purple), DESIGN (lime), CONDUCT (turquoise), COMMUNICATE (orange).

Two additional considerations factored into course design. First, student background in MATLAB programming varied greatly. All students enter the course with some experience with MATLAB programming and applications. However incoming experience ranged as few as two or three weeks of exposure to MATLAB to completion of several upper division courses where MATLAB expertise in solving discipline specific programming problems was developed. Second, we set as a goal a "paperless" classroom, content delivery, and evaluation system to implement the above described course design. Both of these considerations placed additional constraints on course design.

Results:

Specifically, ENGINEERING 1221 is a two credit hour semester course open to undergraduate students upon completion of first year engineering at The Ohio State University or the equivalent. The textbook, *MATLAB: An Introduction with Applications*, 4th Edition[6], covers both the elementary topics introduced in the first year engineering courses and the advanced topics covered in this course, making it an economical choice for the students.
The course content is organized into twelve individual laboratory assignments, four team laboratory assignments and one term project. The content highlights are shown in the figures of the associated poster paper[1]. These figures illustrate the results of the first cycle of the iterative design method. Each figure is framed in the color associated with the learning objective it is meant to illustrate, as described above.

The figures are grouped into the following six categories that depict the major content modules of the course. Each of these major content modules is further described and evaluated in a related paper. References to all of the related poster papers are given below.

MATLAB basics[7], in Figures 1 and 2, illustrates labs that utilize fundamental programming methods such as repetition and selection, user interaction, and array data structures. MATLAB basics applied to wind energy[8], in Figures 3 and 4, demonstrates how MATLAB can be used to solve engineering problems. The board games[9], shown in Figures 5 and 6, give students the opportunity to learn array properties in game application programming. In addition students learn how to modify a MATLAB graphic user interface (GUI) template to suit the user interaction required by their game. The term project[10], Figures 7 and 8, shows the results of students working in teams designing, analyzing and planning the construction of wind farms. Figures 9 and 10 show how MATLAB symbolic math[11] instantly solves all the equations and sets of equations they laboriously solved earlier in this course and in many of their math classes. Finally, MATLAB GUIDE and the "technical challenge"[12] shown in Figures 11 and 12 gives students who are interested in advanced methods the opportunities to develop stand-alone software tools and use these tools to solve engineering problems.

The course is taught in a classroom where each student is provided with an engineering workstation having internet access and network access to engineering software tools such as MATLAB and Microsoft Office. Each student has a Carmen Learning Suite account from Desire2Learn[13]. Almost all course content, activities, submission of team and individual assignment, exams, and evaluations are administered through Carmen. This electronic configuration provides a nearly paperless classroom where students have access to course material and course news, can initiate correspondence, and can submit assignments anywhere, anytime.

Conclusions:

This paper reports results from the first round of course design using the iteration of the iterative, incremental design method and backwards design method. We have found that the combination of these two methods is effective in the initial phase of course development. Once course goals and learning objectives are articulated, assignments, teaching methods, and content can be modified or changed in response to student needs, instructor expertise or interest, and availability of new technology. This allows for flexibility while still maintaining alignment of course goals to ABET outcomes.

We are now in the process of performing the second round of iterative course design. Results of modifications to the course design based on student and staff course evaluations are being studied. Comparative evaluations of expected and experienced student performance are in progress. The cost tradeoffs of new equipment and software products that would transform the classroom from 97% "paperless" to a 100% "paperless" classroom are being considered. The results of these studies and considerations will be available for publication in a follow-on paper.

References:

1. **Duane, JosAnn, Liu, Ye and Maynell, Laurie**. Value Creation in Engineering Education through Goal Directed Course Design. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

2. **The MathWorks**. MathWorks Company. The MathWorks. [Online] [Cited: February 1, 2013.] http://www.mathworks.com/company/.

3. **Larman, Craig and Basil, Victor**. Iterative and Incremental Development: A Brief History. Computer. [Online] June 2003. [Cited: February 1, 2013.] https://www.it.uu.se/edu/course/homepage/acsd/vt08/SE1.pdf.

4. **Wiggins, G P and McTighe, J**. Understanding by Design. s.l. : Pearson Publishing, 2005.

5. **Desire2Learn**. Products. A Suite of Products that Transform the Learning Experience. [Online] [Cited: February 13, 2013.] http://www.desire2learn.com/products/learning-suite/d2l-difference/.

6. **Gilant, Amos**. MATLAB: An Introduction with Applications 4th Edition. Hoboken: John Wiley & Sons, Inc, 2011.

7. **Gao, Kun, et al**. MATLAB Fundamentals, Debugging, and the MATLAB Variable Editor. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

8. **Brown, Thomas, et al.** MATLAB Interactive Programming Basics Applied to Wind Farm Design and Analysis. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

9. **Lui, Ye, et al.** Teaching Array Processing by Programming Interactive Board Games. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

10. **Wagner, Benjamin, et al**. Term Project Illustrating MATLAB Application to Wind Farm Design and Analysis. Retrieved February 4, 2013, from courses-services/asee-papers. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

11. **Gao, Jin, et al**. Teaching Critical, Creative, and Practical Thinking with MATLAB Symbolic Math. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

12. **Patton, Ryan and Duane, JosAnn**. Teaching Iterative, Incremental Problem Solving Methods Using MATLAB GUIDE. EEIC ASEE North Central Section Conference, April 5 and 6, 2013, The Ohio State University, Columbus, OH. [Online] April 5, 2013. [Cited: April 5, 2013.] http://eeic.osu.edu/courses-services/asee-papers.

13. **Desire2Learn**. Products. A Suite of Products that Transform the Learning Experience. [Online] [Cited: February 13, 2013.] http://www.desire2learn.com/products/learning-suite/d2l-difference/.