SE 2AA4 / CS 2ME3 Software Design I

Spencer Smith

August 2, 2017

1 Prerequisites

1. SE 2S03, SE 2XA3

2 Learning Objectives

1. Students should know and understand
   (a) Mathematics used for specification (first order logic, set theory, Parnas tables, etc.)
   (b) Software specification (Module interface specification (Abstract objects, ADTs, Generic), modules with external interaction, finite state machines, descriptive versus operational, UML, etc.)
   (c) Verification (White box, black box, analysis, etc.)
   (d) Functional programming
   (e) Object oriented programming
   (f) Software quality
   (g) Software design principles (correctness, verifiability, etc); information hiding
   (h) Modularization and interface design (assumptions, exceptions, methods, minimal, effective, etc.)
   (i) Design patterns
   (j) git, make, LaTeX, doxygen, pyunit, junit
2. Students should be able to
   (a) Identify reasonable assumptions
   (b) Demonstrate an ability to identify a range of suitable engineering fundamentals (including mathematical techniques) that would be potentially useful for analyzing a technical problem.

(c) Recognize and discusses applicable theory knowledge base

(d) Select appropriate model and methods and identify assumptions and constraints

(e) Estimate outcomes, uncertainties and determine appropriate data to collect

(f) Recognize and follows an engineering design process

(g) Propose solutions for open-ended design problems

(h) Understand the role of the engineer in society, especially in protection of the public and public interest

(i) Show an awareness of the PEO and the role of licensing

3 Mapping to Attributes with their Indicators

Knowledge Base for Engineering	
1. Competence in Mathematics	1a
4. Competence in Specialized Engineering Knowledge	1b, 1c, 1d, 1e
Problem Analysis	
1. Ability to identify reasonable assumptions	2a
2. Ability to identify suitable engineering fundamentals	2b
Investigation	
1. Able to recognize and discuss applicable theory knowledge base	2c
2. Selects appropriate model and methods and identifies assumptions and constraints	2d
3. Estimates outcomes, uncertainties and determines appropriate data to collect	2e
Design	
1. Recognizes and follows an engineering design process	2f
2. Recognizes and follows engineering design principles	1f, 1g, 1h, 1i
3. Proposes solutions to open-ended problems	2g
Use of Engineering Tools	
2. The ability to use modern/state of the art tools	1j
Professionalism	
1. Demonstrates an understanding of the role of the engineer in society	2h
3. Shows an awareness of the PEO and the role of licensing	2i

4 Rubrics

The following learning objectives were not measured:

1. Identify reasonable assumptions 2a
2. Demonstrate an ability to identify a range of suitable engineering fundamentals (including mathematical techniques) that would be potentially useful for analyzing a technical problem. 2b
3. Recognize and discusses applicable theory knowledge base 2c
4. Select appropriate model and methods and identify assumptions and constraints 2d
5. Estimate outcomes, uncertainties and determine appropriate data to collect 2e

The measured learning objectives are summarized on the following pages.

Table 1: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Specification mathematics (first order logic, set theory, Parnas tables, etc) 1a	Unable to read and understand mathematics of specification	Able to read mathematical spec, but cannot write math for a spec	Can read and write mathematical specification	Can read and write mathematical specification, including more advanced mathematics
Exam Questions		36 C	1 B, 20 E	21 C
Num Students	5	32	6	7

Table 2: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

	Expectations			
Topic	**Below**	**Marginal**	**Meets**	**Exceeds**
Software specification 1b	Poor performance on these topics	Okay performance	Average performance	Above average performance
Exam Questions		26 B	22 B, 23 C, 24 D, 25 D, 42 C	43 D
Num Students	17	4	16	12

Table 3: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

	Expectations			
Topic	**Below**	**Marginal**	**Meets**	**Exceeds**
Verification 1c	Poor performance on these topics	Okay performance	Average performance	Above average performance
Exam Questions		8 A, 39 B	9 A, 10 B, 13 C, 14 D, 15 D, 16 B, 17 B, 37 E	38 D, 40 A
Num Students	3	28	15	3

Table 4: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Functional programming 1d	Poor performance on these topics	Okay performance	Average performance	Above average performance
Exam Questions		27 C	3 A, 12 D	35 E, 44 A
Num Students	4	13	11	17

Table 5: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Object oriented programming 1e	Poor performance on these topics	Okay performance	Average performance	Above average performance
Exam Questions		28 D	45 F	45 F
Num Students	10	40	0	0

Table 6: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Software quality 1f	Does not understand meaning of software qualities	Understands meaning of software qualities, but does not understand interaction between qualities	Understand meaning and interaction between qualities	Understands meaning and interaction between qualities and has ideas on how to measure different qualities
Exam Questions		30 A	29 D	45 F
Num Students	9	27	14	0

Table 7: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Software design principles 1g	Does not understand	Can repeat definitions	Can apply principles	Understands reason for and tradeoffs between principles
Exam Questions		6 A, 41 E	7 B	45 F
Num Students	15	21	14	0

Table 8: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Modularization and interface design 1h	Does not understand	Can repeat definitions	Can read and write MIS	Can create an MIS that could be used by someone else to implement a module
Exam Questions		2 B	31 E, 32 C	45 B
Num Students	2	44	4	0

Table 9: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Design patterns 1i	Does not understand	Can repeat definitions	Can identify and use a few patterns	Can identify and use patterns that were not explicitly covered in the course
Exam Questions		34 A	33 B	45 F
Num Students	9	8	33	0

Table 10: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Tools 1j	Cannot use git, Python or Java	Can do basics in git, Python and Java	Very comfortable in git, Python and Java	Uses technology beyond what is covered in the class
Exam Questions		19 E	45 F	45 F
Num Students	9	41	0	0

Table 11: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Follows an engineering design process 2f	Poor performance on these topics	Okay performance	Average performance	Above average performance
Exam Questions		4 A	5 B	45 F
Num Students	3	7	40	0

Table 12: Rubrics
Student work used: Assignment 4 (Battleship)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Open Ended 2g	Overwhelemend by open-ended problem	Trivial attempt to solve	Reasonable attempt to solve in a straight-forward way	Excellent attempt to solve, think abstractly and generally
Assig Performance		Low grade	Average grade	Above average grade (complete bonus)
Num Students	3	4	6	65

Table 13: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

Topic	Expectations			
	Below	**Marginal**	**Meets**	**Exceeds**
Understand the role of the engineer in society 2h	Does not understand	Understands definitions, but cannot apply them to a real world context	Can apply definitions to a real world problem.	Passionate about the importance of the role of engineers in society
Exam Questions		18 A	45 F	45 F
Num Students	5	45	0	0

Table 14: Rubrics
Student work used: Final Exam (See 2aa4 Repo for Exam Details and Questions)

	Expectations			
Topic	**Below**	**Marginal**	**Meets**	**Exceeds**
Show an awareness of the PEO 2i	Not aware	Understands facts	Can apply facts in a new context, to case examples	Passionate about the importance of licensing and the PEO
Exam Questions		11 B	45 F	45 F
Num Students	13	32	0	0

5 Identified areas for continuous improvement

From the Vena Course Report:

- Design patterns should be emphasized to a greater extent next year; the lecture on translating between English and math could be shortened.
- One final exam is easier to translate into measurement data. The assignment grades should be broken down further so that they can be used for more fine grained measurement. Should investigate whether it is possible to have at least one written question on the final exam (TA hours are the main problem with this.)

6 Actions to take for implementation of continuous improvement

From the Vena Course Report:

- The tutorials were not fully utilized. Additional exercises should be added so that the tutorials are more productive. Examples from the previous year can be used
- In the future, discrete math should be required as a pre-req. This recommendation has been through the curriculum committee and is now going to the department.
- Some learning outcomes do not really seeem to fit with 2AA4. These Los were not measured this year. Maybe they should be deleted in the future?

7 Mapping back to indicators

This information is available in Vena.