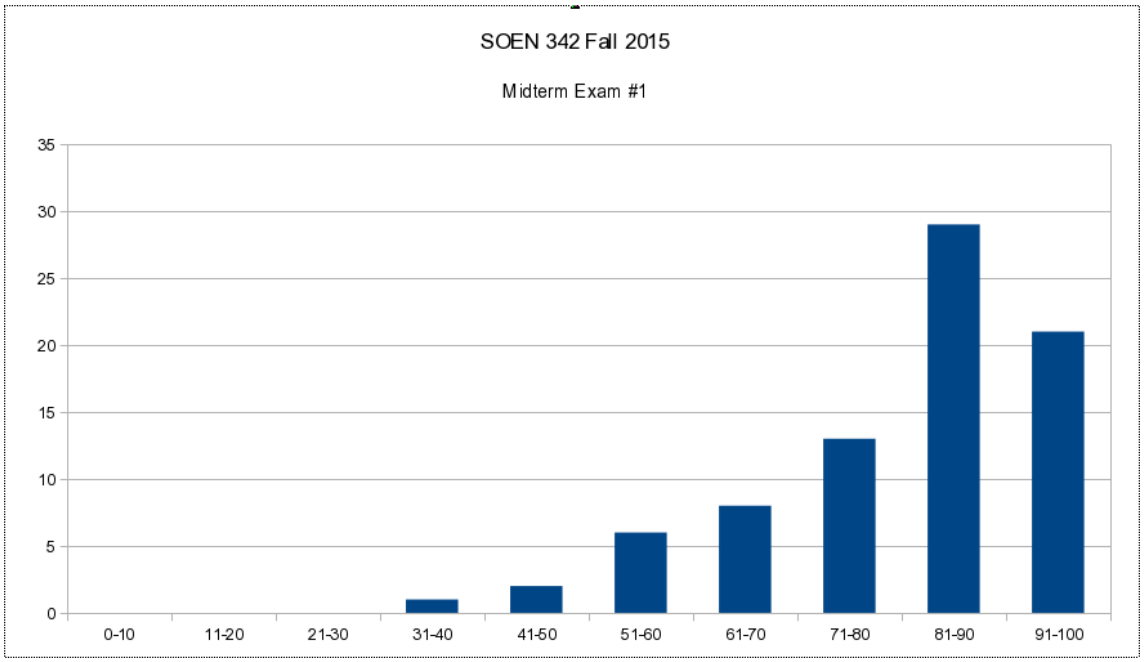
SOEN 342 Software Requirements Specifications

Fall 2015

Midterm Exam I Solutions

Name: _____ Total Points:

ID: _____ _____ / 20



	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Total
Average	3.12	2.81	3.31	0.79	3.83	0.83	7.68	4.51	5.26	32.13
Percent	77.97	93.75	82.66	78.75	76.63	82.50	85.28	90.13	65.78	80.31

Exam Pass Rate: $77/80 = 96.25\%$

(4pts) **1.** Name four different types of *non-functional requirements* (NFRs) and provide a one-sentence definition for each.

4 pts

1. NFR: **Performance**

   Definition: The time required to respond to events or the number of events processed in some interval of time.

2. NFR: **Usability**

   Definition: The ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system.

3. NFR: **Maintainability**

   Definition: The ability to change the system to deal with new technology or to fix defects.

4. NFR: **Security**

   Definition: The system's ability to resist unauthorized attempts at usage and denial of service while still providing its services to legitimate users.

Solution. Above are some examples for correct solutions, but many other NFRs are correct as well.

Discussion. This question is about the classification of requirements into functional and non-functional requirements and their corresponding definitions (covered in Lectures #1 and #6).

Marking. 1 mark for each correct NFR *and* corresponding definition. To qualify as correct, you must have (1) stated an actual NFR and (2) provided the correct definition for it. No marks were given when the definition simply repeated the NFR ("Performance: Requirements about performance", "Security: Requirements about security" and so on). A few provided an *example* for each NFR; but this was not asked: the question is about a *definition* of each NFR.

(3pts) **2.** Provide *brief* answers to the following questions.

3 pts

1. The Unified Process (UP) is one example for a process (lifecycle) model. Name three *other* models for software development:

   (1) **Spiral Model**

   (2) **Waterfall Model**

   (3) **Scrum**

2. Name three different *requirements elicitation* techniques:

   (1) **Interviews**

   (2) **Questionnaires**

   (3) **Group sessions**

Solution. Above are some examples for correct solutions. For other possible answers, please refer to the lectures on process models and elicitation techniques. Note that 'agile' or 'iterative' are not correct solutions to question 1., which is about concrete models, not categories of models.

Marking. $\frac{1}{2}$ marks for each sub-question.

(4pts) **3.** The following questions are about the *Unified Process* (UP). For each question, there is at least one correct answer. There might be multiple correct answers, in that case you need to check all of them to get full marks.

4 pts

Where in the UP can *functional requirements* (FRs) be documented?

- [x] As part of the Supplementary Specification
- [x] As part of the Use Case model
- [ ] As part of the Glossary
- [ ] FRs are not handled in the UP

Based on which criteria are *Use Cases* (UCs) selected for the next *iteration*?

- [x] UCs with the highest risks
- [x] UCs with the highest value
- [ ] Newest UCs
- [ ] Oldest UCs
- [ ] Longest UCs
- [ ] UCs are not handled in iterations

How are *non-functional requirements* (NFRs) handled in the UP?

- [x] NFRs go into the Supplementary Specification
- [ ] NFRs are modeled as Use Cases
- [ ] NFRs are described in the Glossary
- [ ] NFRs are not handled in the UP

What is the outcome of an *iteration* in the UP?

- [x] Tested, executable subset of the final system.
- [ ] Protoype that will be discarded in the next iteration.
- [ ] Requirements specifications (SRS) only
- [ ] There is no defined outcome

Discussion. This question is about the UP (Lecture #02) and the requirements artifacts developed as part of it (Lecture #06).

Marking. For each sub-question, 1 mark for a completely correct answer. $-\frac{1}{2}$ marks for each incorrect or missing checked answer.

(1pt) **4.** Consider the following requirements specification statement:

1 pt

The system must be fast enough to handle peak production throughput.

Do you see any defects in this statement? *(Check only one answer)*

- ☐ This is obviously "Noise"
- ☐ I detect "Inadequacy"
- ☐ Sure: "Poor Modifiability"
- ☑ Yes, "Unmeasurability"
- ☐ No, that's a clear and precise statement.

Discussion. This question targets requirements *defects*, which were discussed in Lectures #1 and #6. The statement is a NFR, but does not contain a fit criterion, hence falls into the "Unmeasurability" defect category.

Marking. 1 mark for the correct answer.

(5pts) **5.** Your task is to document detected requirements overlaps and inconsistencies using an *interaction matrix*.

5 pts

- S1: overlaps (but not inconsistent) with S3, inconsistent with S2 and S4
- S2: inconsistent with S4
- S4: inconsistent with all other requirements

Note: for any requirements pair not defined here, you can assume there is no conflict and no overlap.

(a) (3 pts) Document these overlaps and inconsistencies in the interaction matrix below and compute the values for the total row and column:

Statement	S1	S2	S3	S4	Total
S1	0	1	1000	1	1002
S2	1	0	0	1	2
S3	1000	0	0	1	1001
S4	1	1	1	0	3
Total	1002	2	1001	3	2008

(b) (1 pt) For S2, compute the number of conflicts: **2**

as well as the number of non-conflicting overlaps: **0**

(c) (1 pt) Now compute the *total* number of conflicts: **8**

as well as the *total* number of non-conflicting overlaps: **2**

Solutions. See above (different exams had slightly different rules). Note that in the table: $S_{ij} = 1$: conflict, 0: no overlap, 1000: overlap (but no conflict).

For (b) and (c), you compute the number of conflicts and non-conflicting overlaps as discussed in Lecture #5, by integer division:

- (b) S2: $2 \text{ div } 1000 = 0r2$
- (c) Total: $2008 \text{ div } 1000 = 2r8$

Marking. For (a), 3 point for correct table entries, $-\frac{1}{2}$ marks off for each error. Note that the table must be symmetric. For (b) and (c), 1 mark for correct values ($-\frac{1}{2}$ marks off for each error). If you switched the representation (1000 vs. 1), but did the correct calculations for (b) and (c) based on your table entries, you still received full marks for these questions.

(1pt) **6.** In the quantitative reasoning technique for evaluating options, a Weight(c) is assigned to to each evaluation criterion c and a Score(o, c) is assigned to each option o for c. Then, we can compute the totalScore(o) for an option o as:

1 pt

$$\text{totalScore}(o) = \sum_c (\text{Score}(o, c) \times \text{Weight}(c))$$

Based on the result, which option do you select for a project? *(Check only one answer)*

- ☐ The one with the LOWEST totalScore
- ☐ The one with totalScore (closest to) AVERAGE
- ☑ The one with the HIGHEST totalScore
- ☐ The one with totalScore = 1
- ☐ It's a trick question, I don't use the totalScore for selecting options!

Discussion. This question covers the evaluation of alternatives, which was discussed in Lecture #5.

Marking. 1 mark for the correct answer.

(9pts) **7.** Your task is to prioritize a number of requirements based on the AHP method.

9 pts

(a) (1 pt) You already obtained the relative values for the requirements from your stakeholders. Fill in the remaining entries in the table below:

Objectives	Produce optimal date	Handle preferred locations	Parameterize conflict resolution strategy
Produce optimal date	1	3	5
Handle preferred locations	0.33	1	3
Parameterize conflict resolution strategy	0.2	0.33	1

(b) (3 pts) Now compute the normalized matrix: $R'_{ij} = R_{ij} / \sum_i R_{ij}$ and the relative contribution (last column): $\text{Contrib}(R_i, crit) = \sum_j R'_{ij}/N$

Objectives	Produce optimal date	Handle preferred locations	Parameterize conflict resolution strategy	**Relative value**
Produce optimal date	0.65	0.69	0.56	0.63
Handle preferred locations	0.22	0.23	0.33	0.26
Parameterize conflict resolution strategy	0.13	0.08	0.11	0.11

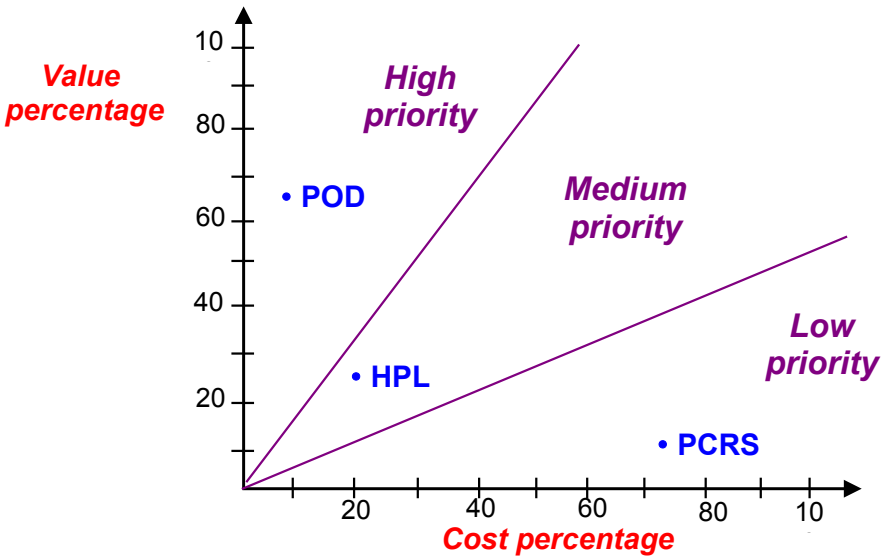
(c) (1 pt) You also obtained the *relative cost* for the requirements from your stakeholders. Fill in the remaining entries in the table below:

Objectives	Produce optimal date	Handle preferred locations	Parameterize conflict resolution strategy
Produce optimal date	1	0.33	0.14
Handle preferred locations	3	1	0.2
Parameterize conflict resolution strategy	7	5	1

(d) (3 pts) Now compute the normalized matrix as before: $R'_{ij} = R_{ij} / \sum_i R_{ij}$ and the relative contribution (last column): $\text{Contrib}(R_i, crit) = \sum_j R'_{ij}/N$

Objectives	Produce optimal date	Handle preferred locations	Parameterize conflict resolution strategy	**Relative cost**
Produce optimal date	0.09	0.05	0.11	0.08
Handle preferred locations	0.27	0.16	0.15	0.19
Parameterize conflict resolution strategy	0.64	0.79	0.74	0.72

(e) (1 pt) Enter the computed values into the cost-value graph below:



Solutions. See above. Note: Different exams had slightly different table values; the solution shown here corresponds to one of these versions.

For (a) and (c), you simply had to insert "1" for the diagonal values, and the inverse $(1/R_{ij})$ for the values across the diagonal. For (b) and (d), you just had to apply the provided formulas to compute the normalized values and relative cost/value columns.

Marking. For (a) and (c), 1 mark for correctly filled-in tables, $-\frac{1}{2}$ marks off for each error. For (b) and (d), 3 marks each for the correct values in the table and the relative cost/value column (again, $-\frac{1}{2}$ marks off for errors in each). For (e), you simply had to plot the three calculated (cost, value) pairs into the graph, for which you received one point, as long as the plot was reasonably close to the computed values.

Discussion. AHP was covered in Lecture #05, a tutorial exercise sheet, as well as the Exercise #3. All formulas were provided, so you essentially only had to understand what the entries in the table mean to correctly assign the diagonal/inverse values.

(5pts) **8.** Consider the following requirements statements:

5 pts

> All applicants have to fill out Form A, except senior citizens, who have to use Form B. However, disabled people of any age, must use Form C instead of A or B. People on social assistance shall complete Form D in addition to any other form. Senior citizens must also complete Form E in addition to the other forms, unless they are on social assistance.

Specify these requirements using a *decision table*:

Senior Citizen?	T	T	T	T	F	F	F	F
Disabled?	T	T	F	F	T	T	F	F
Social Assistance?	T	F	T	F	T	F	T	F
Fill out Form A							X	X
Fill out Form B			X	X				
Fill out Form C	X	X			X	X		
Fill out Form D	X		X		X		X	
Fill out Form E		X		X				

Marking. 1 mark for every correct rule (one row with output actions), $-\frac{1}{2}$ marks off for each error/row.

Discussion. This question addresses requirements specification using decision tables, which was covered in Lecture #6, as well as a tutorial exercise sheet.

(8pts) **9.** You are the requirements engineer in an information system project developing an *Invoicing Orders System* (CBMSys) for the CBM Corp. You already elicited the following information:

8 pts

- The customer requests the CBMSys to place an order.
- CBMSys retrieves customer's credit record from the Customer Persistent Storage (CPS).
- If the customer's credit record is good, then the CBMSys creates a new order and sends the order to the publisher.
- The CBMSys receives the order invoice from the publisher.
- If the order invoice's details are approved by the customer, then the CBMSys accepts it and sends it to the Account Payable System (APS).
- CBMSys generates a sale invoice for the customer.
- The customer provides payment information to the CBMSys, and the CBMSys sends it to the Accounts Receivable System (ARS).
- The CBMSys prepares publisher payment and sends the check to the publisher.
- The CBMSys assigns shipment to the order.
- The customer shall be able to view the order status from the CBMSys.
- The customer enters the order ID to the CBMSys, and CBMSys shows the order status to the customer.
- Customer and publisher shall be able to update their Personal Profile with the CBMSys.

(a) (2 pts) Identify 4 *actors* and give a brief description (3–5 words) for each

- **Actor 1:** Name: **Customer**

  Description: Main user, triggers order invoices
- **Actor 2:** Name: **Publisher**

  Description: Receives orders, handles invoices and payments
- **Actor 3:** Name: **CPS**

  Description: Supplies credit records to CBMSys
- **Actor 4:** Name: **ARS**

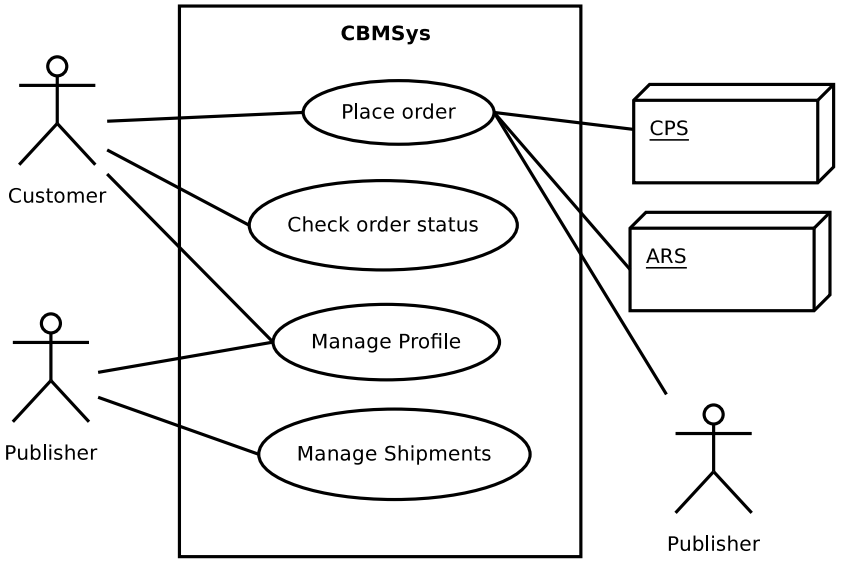
  Description: Receives customer's payment information

⇒ ***Continued on next page!***

(b) (4 pts) Identify four primary (user-goal level) use cases and related actors (identified by A1, A2 etc.):

- **UC1** Name: Place order Actors: A1, A2, A3, A4
- **UC2** Name: Manage Profile Actors: A1, A2
- **UC3** Name: Check order status Actors: A1
- **UC4** Name: Manage Shipments Actors: A2

(c) (2 pts) Draw the UML use case context diagram for your actors and use cases:



Solutions. See above for a possible solution, but many other variations are possible (in particular, including the APS actor).

Marking. For (a), $-\frac{1}{2}$ marks for each incorrect actor (both name and definition must be correct and consistent). In particular, it is a mistake to model the system that is being specified (CBMSys) as an actor for itself.

For (b), 1 mark for each correct use case name and corresponding actor list, which must correctly reference the actors defined in (a). No marks if your use case was not user-goal level as asked (e.g., use-case steps or sub-function level use cases, like *login* or *pay invoice*). Use case names should be written as *active verb phrases*, but no marks were taken off for this if your use case was otherwise correct. For (c) you had to translate the use case model from (b) into a use case context diagram. Marks were taken off ($-\frac{1}{2}$ each) if you did not show all use case/actor relations defined in (b). Note that each use case must have a primary actor (shown on the left in the UC context diagram), whereas secondary actors (shown on the right in the diagram) are optional.

Discussion. This question targets requirements specification with use cases, which was extensively covered in Lecture #7. The main goal here was to demonstrate that you can

synthesize a coherent systems and model it based on the use case methodology, starting from the individual requirements statements that were provided (in practice, these could have come from many different elicitation sources).

— End of Exam Solutions —