DIT046 / DAT356 Requirements and User Experience
Final Exam

August 18, 2022

Examiner/Contact Person

Jennifer Horkoff
Jennifer will come to the room to check for questions at roughly 15 and 17.
Alternatively Phone: 0733 050517

Authorized Aids

Textbook: Designing Interfaces, 2nd or 3rd Edition, By Jenifer Tidwell (Charles Brewer & Aynne Valencia)

Grading Scale for Exams

| % Grade	Final Grade
0-49	Fail (U)
50-64	3
65-79	4
80-100	5

PLEASE OBSERVE THE FOLLOWING:

- All answers must be in English. Answers must be legible and readable.

- Sort the questions in order before handing them in.

- Put the number of the question on every paper.

- This exam has 7 pages.

Part 1: Multiple item and Short-Answer Questions

Question 1: Multiple Choice Questions (10 points)

On your paper, write the name of the question, then the letter(s) of your chosen answer(s), e.g., 1.1: c., 1.3: a, d, e. For each question, there may be more than one correct answer. For full marks, list all the correct answers. Each correct answer listed is +1 point, each incorrect answer is -1 point.

1. Which of these requirements captures design information or unnecessary constraints and is therefore not well-formed? Note: the examples are about lifts, also known as elevators. In this question there may be more than one correct answer, list them all.

   (a) A lift will only reverse direction when stopped at a floor
   (b) The system will cycle the lift doors every time that a lift stops at a floor
   (c) The lift buttons must use only red and green as identifying colors
   (d) The lift must communicate with a central server every 30 seconds
   (e) The lift must never be moved with the doors open
   (f) The lift class shall have a "date installed" attribute
   (g) Each lift should be used an approximately equal amount

2. Which UI design patterns are implemented in the following UI? There may be more than one.
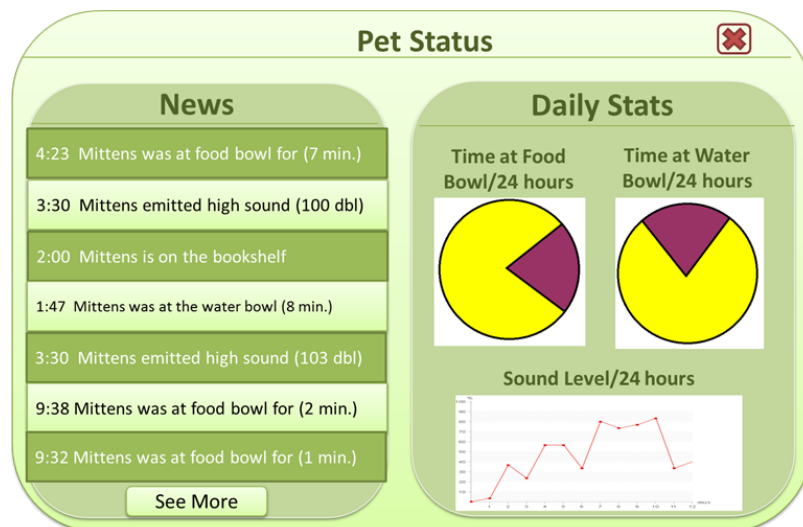
   (a) Escape hatch
   (b) Feature, search and browse
   (c) News stream
   (d) Sharing widget
   (e) Other people's advice
   (f) Titled sections
   (g) Center stage

3. Which of the following statements about the models/diagrams we have covered in the course are true? There may be more than one correct answer.

   (a) Goal models are used to understand actors and dependencies between actors

(b) Customer journey maps and are used to understand qualities and contributions

(c) Use case models are used to understand actors and inputs and outputs

(d) Context diagrams are used to understand actors and inputs and outputs

(e) Goal models are used to understand qualities and contributions

(f) Use case models are used to explore scenarios

(g) Customer journey maps and are used to explore scenarios

Question 2: Short-Answer Questions (20 points)

1. What is the difference between a domain property/assumption and a requirement? Between a constraint and a requirement? Define domain property and constraint, and state how they relate to requirements. (4 marks)

2. We discussed the desired qualities of SRS requirements vs. the desired qualities of user stories. Explain how the desired qualities are the same or different. List two similarities and two differences and explain why they are similar or different (4 points)

3. Describe exploratory, combinatorial and transformational creativity. The explanations should make it clear how these types of creativity are different. (6 points)

4. Why is user experience (UX) important? List four reasons. (4 marks)

5. What is a minimal viable product? Why is it useful? (2 marks)

Part 2: Domain Example and Long-Answer Questions

The remaining questions on the exam will relate to a problem in a domain, as described below. The scenario describes the situation today and the expectations for the new system. Focus your analysis and modeling on the envisioned (to be) system, but keep in mind the problems and requests with the as is situation, trying to avoid problems and satisfy user needs.

Newspaper Planning

Planning articles in a newspaper editorial office The software house BIT develops IT products for companies that publish newspapers and other periodicals. BIT is now developing a software product that can help the newspaper editors plan the contents of the daily newspaper and manage the work of the journalists.

Editorial Board Meeting An editor collects ideas for articles all the time. The ideas come from the newspaper's journalists and from the editor himself. Each morning the editorial board (the editor and some of the journalists) gathers to plan the contents of tomorrow's newspaper. They look at the list of ideas, present their own ideas, etc. The result is a list of articles to go into tomorrow's newspaper. There may also be suggestions for articles to appear later, for instance because it will take more time to write the article.

Work Allocation Next, they allocate the work. Each article gets a responsible journalist and often a photographer too. Each of them has to perform a job (also called a task) that is part of the article production, and they coordinate with each other as a team. During the planning, the board looks at who is good at making this article. All the time the board pays attention to what other jobs the employees have and how busy they are.

The editorial board has a good sense of who is good at writing what, how long the article will be, and how many articles are needed for tomorrow's newspaper. They don't need support for this. However, it is hard to get an overview of who writes what, and whether it is too much for the individual journalist. The software system must help them here.

Reading and Updating the Plan When the plan is ready, journalists and photographers are told what to do. Today this is done by phone or by email. The idea with the new system is that they are told automatically when their plan has been made or changed. They can then see their own plan - including comments. Initially they have to indicate whether they accept the job. Later they have to report that it is finished. During the work, it may turn out that it cannot be finished today, but has to wait for tomorrow. It may also happen that the article has to be abandoned entirely, for instance because there wasn't a good story to tell after all. Such changes are of course very important to report to the editorial board via the system.

Vision (system to be) You will write requirements and perform UX design for a new editorial work planning system. The system should collect ideas, articles, assigned roles for each day, and for future days. This information is updated daily at the editorial board meeting. The plan should be changeable by all involved, with all involved parties notified. There should be warnings if particular roles (journalists, photographers) are overbooked. If a number of articles are late or cancelled, the editorial board needs to know, else the paper for the next day may be lacking content. Security is an issue, as journalists may cover important stories with sensitive content. The systems should account for all applicable privacy laws.

You can assume that all roles have access to standard PCs, the internet, and phones. They may use the system both in the office, and out on location. They would like the system to be finished by January of 2025.

Question 3: Context Diagram (10 points)

For the given case, draw a context diagram. Add the system actor, identify the relevant stakeholder/actor entities, and the relationships between the system and these entities, labelled with high-level inputs and outputs. Try to capture at least four actors (the system actor counts), and appropriate relationships. Supplement the diagram with text to explain any ambiguous or unclear parts of the model.

Question 4: Use Case Diagram (10 points)

For the given case case, draw a use case diagram. Capture the system and other relevant stakeholders. Identify the major use cases and relationships, both between the stakeholders and the use cases, and between use cases. Try to capture three actors, and appropriate use cases.

Question 5: Goal Model (10 points)

For the given case, draw a goal model. Identify the relevant actors, including the system actor. Try to capture four actors, and important dependencies between actors. Show the internal goals/tasks/resources/qualities for at least one actor, the others can be "closed" showing only dependencies to and from the actors. See the legend below for open and closed actors. For the open actor, capture the desired goals, tasks, resources, and qualities. Add some internal relationships between these elements.

There are many elements that can be included in the model. Aim for: four actors including one open actor with an actor boundary, at least one goal, quality, resource, task, four dependency links, two contribution links and two AND/OR links. In other words, the model does not have to be complete as per the domain description. It's helpful to use the information already captured in the other diagrams; however, we won't mark you on consistency between diagrams. Supplement the diagram with text to explain any ambiguous or unclear parts of the model.

Goal Model Legend (note, can use alternative shapes and labels as long as the mapping to the original concepts is clear)
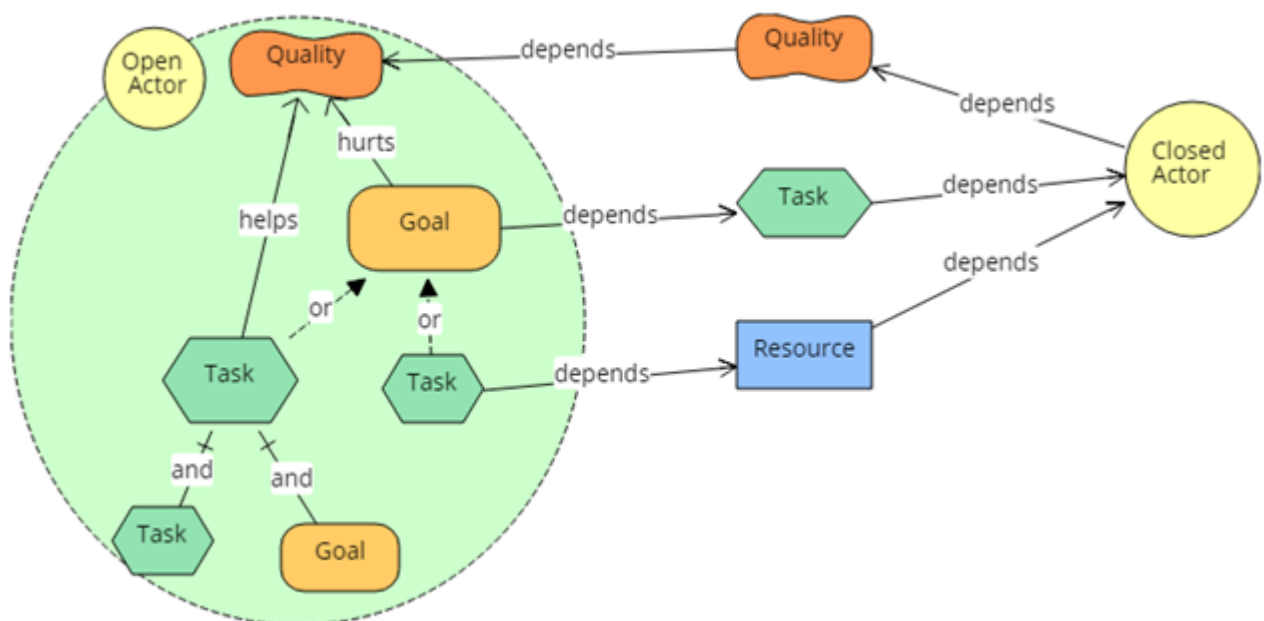
Question 6:   Textual Requirements (15 points)

Record requirements for the case provided. List four functional requirements in SRS form, two non-functional requirements in SRS form, two domain assumptions, and two constraints. In addition, list five user stories. The user stories should be unique, i.e. not a repeat of one of the SRS requirements. Remember the desired characteristics of user stories and SRS requirements when recording your requirements.

Question 7:   UI Design & Patterns (20 points)

Draw two UI screens for the given case implementation. Screens can include pop-up windows, just be clear how and why they appear. Each screen should use at least two different patterns from either Tidwell book, four different patterns in total. Each screen should implement at least two functional requirements (they do not have to be the requirements from question 6, just make it clear what requirements the screens implement). For each screen, write: the two requirements implemented (and how, if not obvious), the two patterns implemented, and why the patterns are a good choice for the screen and its functions.

Question 8:   User Testing Tasks (5 points)

Come up with a list of five test tasks for the users to perform during usability tests for the two designed screens. Each screen should be tested by at least one task, for at least two tasks, indicate which of your two screens they test. The tasks should cover some of the main use cases or requirements of your system. Tasks should conform to the desirable qualities of user test tasks as discussed in the lecture.