DIT045 / DAT355 Requirements and User Experience
Final Exam

April 9, 2021, 14:00

Examiner/Contact Person
Jennifer Horkoff: jenho@chalmers.se / contact via Canvas
Email / message Jennifer via Canvas. She will check email at 14:30, 16:00, 17:30

Authorized Aids
Textbook: Designing Interfaces, 2nd or 3rd Edition, Patterns for Effective Interaction Design, By Jenifer Tidwell
Course slides and assignments

Note: you have an open book exam and can access the internet but accessing random examples or information on the web is not recommended. You may find examples or advice which differs from course material.

Note: although the exam is open book, you are not allowed to collaborate with each other when creating answers. Any written or drawn answer which is very similar to another will be reported for plagiarism.

See announcement for detailed information on how to prepare and submit your file:
https://chalmers.instructure.com/courses/14831/discussion_topics/62247

GU Grading Scale for Exams

% Grade	GU Grading Scale
0-49%	Fail (U)
50-75%	Pass (G)
76-100%	Pass with Distinction (VG)

Chalmers Grading Scale for Exams

% Grade	Chalmers Grading Scale
0-49%	Fail
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.
- Start each long answer on a new paper.
- Sort the questions in order before handing them in.
- Write your student code on the first page and put the number of the question on every paper.

Part 1: Multiple Choice 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.

1.1 Which of these requirements captures design information and is therefore **not** well-formed? The requirements are about an application which monitors pets. In this question there may be more than one correct answer, list them all.

   a) The system shall allow for monitoring pet eating, it will monitor the length of time that a pet is within 20 cm of their food bowl.
   b) The system shall store all pet eating time information in a database with a minimum size of 2 GB.
   c) The system shall allow users to video chat with pet using an iPhone 9.
   d) Only authorized users will be allowed to video chat with pet.
   e) Upon starting chat, the system will play a sound to attract pet, the sound will be a loud beep.
   f) Video chat will include sound in both directions.
   g) The system will be designed using a service-based architecture.
   h) The system shall monitor temperature, the current temperature can be displayed to the user
   i) The temperature reading will be accurate within 5 minutes.

1.2 Which UI design patterns are implemented in the following UI? There may be more than one.

   a) Escape hatch
   b) Sharing widget
   c) Titled sections
   d) Row striping
   e) Thumbnail and Text List
   f) Infinite List
   g) Feature, search and browse

1.4 Which of the following statements is correct?   One or more may be correct.
   a) Domain properties are requirements, they are requirements that come from the domain.
   b) Domain properties are not requirements, they do not have to be satisfied by the system.
   c) Users are stakeholders, not all stakeholders are users.
   d) Stakeholders are users, not all users are stakeholders.
   e) Constraints are not requirements, they are properties of the domain.
   f) Constraints are requirements, they are requirements that do not come from user needs.

Question 2: Short-Answer Questions (15 points)

2.2 User testing can be used from both a requirement engineering (RE) and user experience (UX) perspective. Explain how user testing can be useful for RE and how it can be useful for UX, in other words how do the aims and benefits differ from each perspective. Give two benefits from each perspective. (4 marks).

2.3 What are some differences and similarities between context diagrams and use cases? Name two similarities and two differences (4 marks).

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

2.3 What is wrong with the following SRS requirements? Keeping in mind the desired qualities of requirements, list three quality issues found in the requirements below. (3 marks)

The pet monitor system should allow pet owners to turn on their pet-monitor camera at any time.
The pet monitor system should be able to predict the future needs of all pets.
Pets monitored by the pet monitor system should be happy.
The pet monitor system should only allow the camera to be used when the room has sufficient light (>10 Lux).

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 with the way of working, trying to avoid some of the issues reported.

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, e.g. the 7th April edition. 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 must 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 personally, by phone or by email. The idea with the new system is that they are told automatically by email or SMS when their plan has been made or changed. They can then see their own plan - including comments. Initially they must reply whether they accept the job. Later they must report that it is finished. It may turn out that a job cannot be finished today but must wait for tomorrow. It may also happen that an article must be abandoned entirely, for instance because there wasn't a good story to tell. 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 updatable by all involved, with all involved parties notified. There should be warnings if particular roles (journalists, photographers) are overbooked. If several 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 2023.

Question 3: Use Case Diagram (10 points)

For the newspaper planning system, 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 4 actors, and at least 5 appropriate use cases.

Question 4: Scenario (15 points)

Pick one use case from Question 3. For this use case, fill out Cockburn's Use Case template (obligatory template elements listed below).   Flesh out the details of the process, including exception cases and error cases. Pick a use case with at least one extension and sub-variation. Consider preconditions.

Cockburn's Use Case Template (reduced)

Use Case: <the name should be the goal as a short active verb phrase>

CHARACTERISTIC INFORMATION
Goal in Context: <a longer statement of the goal, if needed>
Scope: <what system is being considered black-box under design>
Preconditions: <what we expect is already the state of the world>
Success End Condition: <the state of the world upon successful completion>
Failed End Condition: <the state of the world if goal abandoned>
Primary Actor: <a role name for the primary actor, or description>
Trigger: <the action upon the system that starts the use case, may be time event>

MAIN SUCCESS SCENARIO
<put here the steps of the scenario from trigger to goal delivery, and any cleanup after>
<step #> <action description>

EXTENSIONS
<put here the extensions, one at a time, each referring to the step of the main scenario>
<step altered> <condition> : <action or sub.use case>
<step altered> <condition> : <action or sub.use case>

SUB-VARIATIONS
<put here the sub-variations that will cause eventual bifurcation in the scenario>
<step or variation # > <list of sub-variations>
<step or variation # > <list of sub-variations>

Question 4: Goal Model (15 points)

For the newspaper planning system, draw a goal model. Try to capture 3 actors, including the system actor, and appropriate elements and dependencies. Show the internal goals/tasks/resources/qualities for one "open" actor, the others can be "closed" showing only dependencies to and from the actors. See the legend below for open and closed actors. There are many elements that can be included in the model, aim for: 3 actors (1 open actor, 2 closed), 6 elements (which include at least one goal, quality, resource, and task), and 6 links (2 dependency links, 2 contribution links and 2 AND/OR links). The model does not have to be complete as per the domain description.

It's helpful to use the information already captured in the use case diagram to draw the goal model diagram; 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)
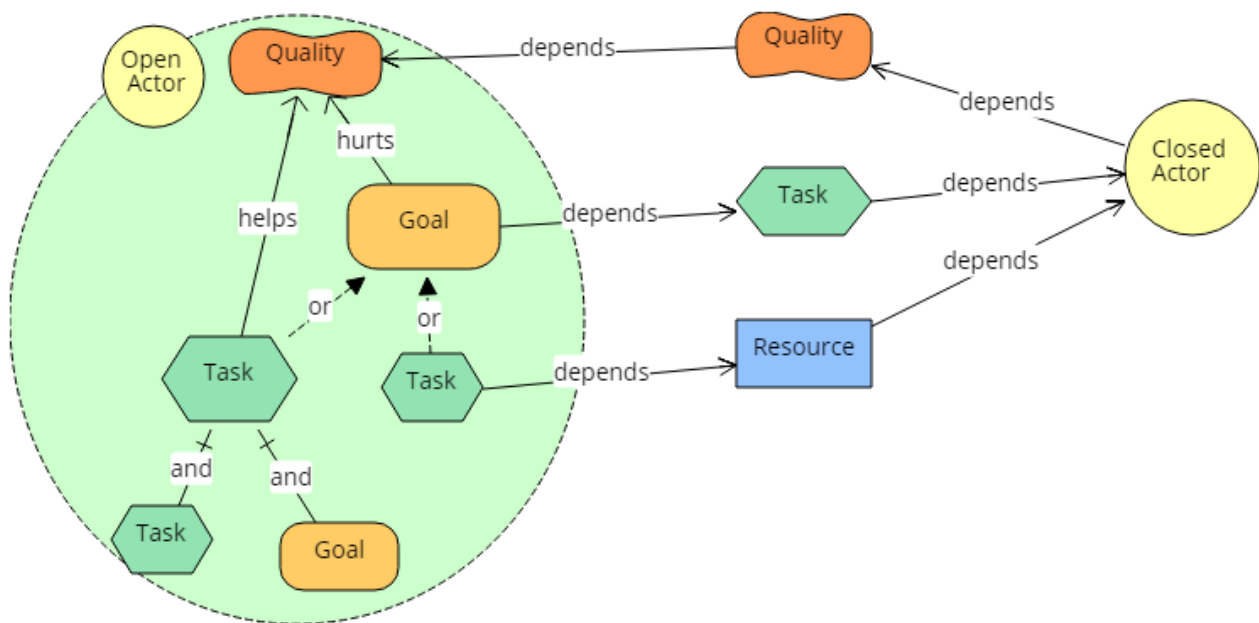
Question 5: Textual Requirements (15 points)

Record requirements for the newspaper planning system. 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 (either functional or non-functional). 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 6: UI Design & Patterns (20 points)

Draw two UI screens for the newspaper planning system vision. Screens can include pop-up windows, just be clear how and why they appear. Each screen should use at least two different patterns from the Tidwell book (2nd or 3rd edition), four different patterns in total. Each screen should implement at least two functional requirements (they do not have to be the requirements from the previous question, just make it clear what requirements the screens implement).   The requirements implemented can be SRS requirements or user stories, just ensure they are functional and not NFRs.

For each screen, write: the two requirements implemented (and how, if not obvious), the two patterns implemented, where they are implemented, and why the patterns are a good choice for the screen and its functions. Hint: the screens do not have to be complicated; it is more important that they correctly implement patterns and requirements and we can understand what they do.

Question 7: 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 used in at least one task. Indicate which, if either, screen is tested by a task (e.g., screen 1, screen 2). The tasks should cover some of the main requirements of your system which are implemented in the two screens. Tasks should conform to the desirable qualities of user test tasks as discussed in the lecture.