Faculty of Science, Engineering and Technology

User-Centred Design

Task 7.4 Usability Evaluation Tasks

ILO	Apply evidence-based approach to requirements elicitation, specification and evaluation.
Purpose:	Part 1: Tasks are a critical component of a usability evaluation. A task gives the user something to do in the evaluation. Badly written task will completely invalidate your usability evaluation. Learn how to write tasks for your user to do during your evaluation. Part 2: The fact that someone could click through the tasks in your interface does demonstrate that you have met functional requirements. However, we are also interested in whether the real users can do it. Usability requirements help us set targets for user performance.
Group Task:	Write tasks for the user to complete in the usability evaluation. Write usability requirements for your tasks.
Resources:	■ *Lecture Notes:* Topic 12 User Evaluation Method ■ *Textbook:* Hartson & Pyla Chapter 12, 14 and 15 or Stone et al. Chapter 22, Snyder (2003) Paper Prototyping, Chapter 6 (on writing tasks for usability evaluations) ■ *Template:* UCD Task Descriptions.docx
Deliverables	Part 1: Task descriptions Tasks that test the usability of your prototype. Part 2: Usability requirements
Marking Criteria	Part 1: Task Descriptions: Tasks have the following characteristics: ■ gives the user a task with a clear end point ■ does not instruct the user on how to use the interface ■ does not bias participants in any way ■ does not require the user to use their personal details Part 2: Usability Requirements Usability requirements have the following characteristics ■ Effectiveness, efficiency and satisfaction requirements are provided for each task ■ user group/role is specified ■ measuring instrument is described for each requirement ■ metric is provided for each requirement ■ target estimates are provided for each metric

Instructions

Part 1: Task Descriptions

To test your prototype you will need to get your participants to do some tasks. These tasks should reflect the core functional requirements of your design.

The way the tasks are worded is critical to the success of your evaluation. Make sure that your tasks:

- Give the user something concrete to do
- Have clear completion criteria (i.e., book appointment for 13/04/2015 at 11:30)
- Do not give instructions (or hints) about how to use the interface

1. Write some tasks for your usability evaluation. Aim to give your user about 5 minutes worth of things to do. The tasks should reflect those covered in your design scenarios.

Hint: Make sure that your prototype will allow participants to do these tasks! Pilot testing of tasks on your prototype is highly recommended before you start testing!!!

Usability Task Exercise: Which Example task is best suited for usability testing?

Example A

Train Tracker is very easy to use. Use Train Tracker to find the next train.

Example B

Step 1: Type the name of the station you are leaving from into the Departure field

Step 2: Type the name of the station you want to go to in the Arrival field

Step 3: Tick the “Leaving in next 10 minutes” box on the left

Step 4: Click on the search button

Step 5: A list of trains leaving in the next 10 minutes will be displayed (Note: earliest time at top).

Step 6: Write down the time of the first train to leave: _____

Example C

Use Train Tracker to find the next train going to Frankston from Flinders Street.

Time of next train to Frankston: _____

Suggested answer available in Appendix A

Part 2: Usability Requirements

Hopefully, if your prototype has been constructed properly, your tasks reflect the functional requirements of the software and it will be 'theoretically' possible to complete the tasks provided in Part 1 of this Task. If it can, then you have met the functional (user) requirements for the software.

However, that does not mean we are finished and can now go home! It is also important that the user can actually work out how to do the task and/or does not take an unreasonable amount of time to do it. If they can't find or use a function, it might as well not be there at all. Also it would be nice if they felt happy and comfortable using the software!

In Part 2 of this task you will set some expectations about how well you expect the users to do the tasks with your interface. This type of requirement is known as a usability requirement) and the focus is on quality of the interaction. Usability requirements are a type of non-functional requirement.

A good usability requirement specifies the following things:

- context of use (e.g., users, goals and environment)
- the method you are using to assess the requirement (e.g., performance on a task or rating on a questionnaire)
- what you are measuring (aka the metric) (e.g., task completion, time taken to complete task, difficulty rating etc)
- target performance - what level of performance is acceptable for the function to be considered usable

See box below for an example of a usability requirements table that summarises the requirements for a gym exercise app.

1. Construct a table showing usability requirements for effectiveness, efficiency and satisfaction for each of your tasks.

One nontrivial issue with writing usability requirements is coming up with an estimate of the performance target. There are three ways of establishing target values:

1. Business case (when the client has operational requirements – e.g., task must be completed in 5 sec or system not worth implementing)
2. Competitive testing (user testing competitor products – e.g., users can complete task in 10 sec on competitor, we need to be at least as good)
3. Expert testing (for prototype interfaces, or when there is no business case or competitor available – e.g., use completion time for product expert (i.e., designer/team member) used to estimate how long a new user would complete the task)

2. Specify target values for your usability requirements. OPTIONAL: Use competitive or expert testing to determine target values for your usability requirements. Otherwise indicate in your answer that you have made a guess to estimate the targets.

Example: *Gym exercise recording app for iPhone*
The following is an example of some usability requirements (including target values) for a Gym exercise recording app.

User Role	Usability Goal	Measuring Instrument	Metric	Target
Current Gym member	Effectivness	Task: Imagine you have just completed three sets of bicep curls with 5 kg weights. In the first set you managed 15 reps, in the second 14 reps, and in the third 12 reps. Update the app with this information.	Proportion of users who complete task without assistance	80%
Current Gym member	Efficiency	Task: Imagine you have just completed three sets of bicep curls with 5 kg weights. In the first set you managed 15 reps, in the second 14 reps, and in the third 12 reps. Update the app with this information.	Average unassisted task completion time (sec)	10 sec
Current Gym member	Satisfaction	Task: Imagine you have just completed three sets of bicep curls with 5 kg weights. In the first set you managed 15 reps, in the second 14 reps, and in the third 12 reps. Update the app with this information.	Average task difficulty rating (1 = very easy, 5 = very hard)	2.5 or less
Current Gym member	Satisfaction	System Usability Scale (SUS)	Average SUS score	68 or higher

These usability requirements constitute a promise to the client on system performance. The purpose of a summative usability test is to demonstrate that these performance metrics have been met.

Appendix A: Answers to Review Questions

Usability Task Exercise: *Which Example task is best suited for usability testing?*

Example A

Train Tracker is very easy to use. Use Train Tracker to find the next train.

Answer:

Not this one because :

Bias: It tells users that app is easy to use, this may influence how they perceive the app an interpret their interaction with it.

No clear completion point: Sometimes it is interesting to allow users to explore an app with no particular task in mind. However, in most formal usability testing we want to know if they have been able to complete a task or not. Giving them a realistic task is the easiest way to do this (although some researchers will allow users to define the task - but this makes generalisations across users)

Example B

Step 1: Type the name of the station you are leaving from into the Departure field

Step 2: Type the name of the station you want to go to in the Arrival field

Step 3: Tick the "Leaving in next 10 minutes" box on the left

Step 4: Click on the search button

Step 5: A list of trains leaving in the next 10 minutes will be displayed (Note: earliest time at top).

Step 6: Write down the time of the first train to leave: _____

Answer:

Not this one because :

Bias: It tells users how to use the app (e.g., which options to select and in which order). This completely negates the purpose of the usability evaluation which is to see if the users can work out how to do the task without instruction.

Note: You can still provide Help as part of the app or documentation that goes with the app, but not in the instructions for the usability evaluation.

Example C

Use Train Tracker to find the next train going to Frankston from Flinders Street.

Time of next train to Frankston: _____

Answer:

This one because :

No Bias: It does not tell the user how to use the app (e.g., which options to select and in which order). Or give them any clues (e.g., but using terms used in interface like Departure and Arrival).

Clear Task: There is a defined objective for the task, the user can be successful or make errors in task completion. It also means the user does not have to spend time thinking of a task to do, so task time reflects time working on interface and not something else.