Task A7.2C: finding sunny users

Running usability tests

Due: by Monday of week 10, for review ahead of week 10's lab

Aim

The aim of this task is for you to demonstrate your ability at running a short usability test.

You should have submitted A7.1P prior to starting on this task.

Tasks

1. Usability testing of prototype

You must perform a usability test on the prototype of the Suntime app that you created in Assignment A7.1P.

The usability test must be undertaken **with at least 3 people** (that are not enrolled in this unit).

The usability test method (read all steps carefully):

1. Provide each tester information about the application: Idea, Motivation, Key Features and Scenarios (see details provided in previous assignments). The tester should read all of the information first.
2. Ask each tester to perform all of the scenarios as described using the prototype and ask for feedback
3. While they are undertaking the test ask them to say out loud what they are thinking (or) if they are currently not clear how they should proceed -- record these comments. While they are undertaking the test, you are not allowed to offer clarifications around the UI even if they are stuck/confused. You are however allowed to act as the phone's sensor and provide them with information such as GPS location.
4. If they are unable to complete a scenario, record that as such (including additional information on why the tester was stuck).

At the end of the test, ask users to rate the following aspects on a 5-point scale [where, 1 is poor and 5 is excellent] as well as a brief comment for each item:

- Clarity of the UI in communicating how to complete the scenario (i.e. tester is able to use the correct set of features to complete the task without any assistance);
- The visual clarity of the sketch/prototype (specifically, readability);
- The effectiveness of the overall layout of components and functionality;
- General difficulty in using the application; and
- Effectiveness of the prototype with respect to communicating the intent of the application.

Short Report on Usability:

Once the usability test is complete, summarise the findings in a Short Report. Your usability short report must contain the following sections:

1. Introduction (What does this report contain? Describe the task briefly, why you under-took it and key findings)
2. Usability Test Method (Describe the method -- rephrase above method and adjust if you changed it slightly)
3. Findings a. General comments from the users (while performing scenarios) b. Survey results (conducted at the end of the test)
4. A short report aims to capture your understanding of a topic area. It should include at least 1 or 2 references to support the points you are making. The report is expected to be between 500 - 1000 words.
5. Discussion on Usability test (What was surprising/interesting/unexpected? What areas need attention? What did you learn by doing the usability test? What would you change in your design?)
6. Summary (Key findings from usability test, A short summary of your reflection)
7. References
8. Appendix (Survey forms, results and notes should be placed here)

What should you do if run out of space (given the 1000 word hard limit)?

Shorten the usability test method section by citing the description in this handout.

Note: Tables and Images are counted as 200 words each (independent of their size).

Checklist

- ○ Usability test conducted as per specification
- ○ Usability test reported using the format suggested
- ○ Survey information presented in Appendix

Tasks for COS80019 students only

The following tasks are core tasks that only Masters students should attempt. These tasks are optional for undergraduate students. The general concepts related to these questions are covered in the lectures, but Masters students are also expected to read, briefly research a bit more broadly to answer these questions.

2. Navigation flow modelling

Draw a navigation model (using the notation presented in the lectures) for one of the following apps: IMDb, LinkedIn, Eventbrite, AFL, Facebook, Dictionary.com, any Reddit client, any internet banking apps, Carsales.com, WhatsApp, Dropbox, Evernote, Realestate.com.au (or any other app of your choice that is not a game and has some complexity in UI).

Note: All apps in the list have a similar complexity, so pick a random one (to reduce time spent deciding which one).

- You need to model at least 4 - 6 activities.
- The model must capture global and contextual navigation elements.
- The model must identify any dialogs used within the application
- The navigation model must be supported by screen shots of the app as well.

In the submitted report include the navigation model, screen shots of app, and present an argument for the value of navigation models (over sketches). Also, comment on the limitations and gaps in the navigation models. The write up needs to be less than a page.

Core/Extension Tasks

All tasks in this assignment are “extension”. You must complete all extension tasks, submit for feedback, and achieve a pass for all tasks in order to be eligible for a credit or higher grade in this unit.

Submission

You are required to submit a PDF report using Doubtfire:

- login to Doubtfire at http://doubtfire.ict.swin.edu.au
- The header (or) footer of the document must contain your name, student id, and unit code.
- The document must have a title (e.g. Submission for Assignment <number>)
- Evidence that shows you completed each task must be presented in a separate section.
- The document does NOT need a table of contents nor a cover page.

The reports are assessed and feedback given via Doubtfire and, if required, in your lab. You are expected to incorporate the feedback (esp. if changes are required) and submit the changed reports as part of the final portfolio.

Note: This is a formative assignment. That is, an assignment designed to provide feedback. If you fail this assignment, you have one week to make corrections and resubmit to pass.

Demonstration

You may be asked to demonstrate your assignment in the lab. You should be able to do this and explain your code when asked in the lab session.