COSC368 2021

Assignment: COVID-19 Edition Design of a Flatting App

Due: 5pm Wed, 20 Oct., 2021
Late submissions -15, until 5pm 27 Oct.
Worth: 30%

This assignment will help you develop critical skills in interaction design:

- identifying user needs and analysing tasks;
- sketch-based generation of design alternatives;
- assessing the relative merits of design alternatives;
- design refinement and iteration;
- user interface specification;
- written communication of interaction needs, user tasks, UI specification, and design rationale.

Background

FlatMate Ltd. is a startup company that wants to develop software tools to help flatmates manage their flat. They have contracted the famous interaction design consultancy YOUR TEAM NAME HERE to develop a design concept for their mobile app. They are interested to learn what tasks users might want to complete through the app, and they wish to see evidence of your iterative design from initial design concepts to a final unambiguous UI specification.

Design brief

Your design brief from FlatMate Ltd.'s CEO is as follows:

"*We want to develop a set of software tools that will become indispensable for managing and coordinating flat activities, helping to keep it happy and harmonious. We want you to design an interface that will give flatmates access to handy utilities that ease running the flat. We want to know what things flatmates will want to do using your interface; we want to see your sketch-based initial design concepts; and we want to see how these sketch-based design concepts were iterated upon to produce your final unambiguous user interface specification.*

Your design rationale is important to us too. We want to know why you think the set of tasks and activities you identify are important to users; we want to know how you see the pros and cons of your designs (both sketches and final UI specification).

We understand that supporting some flat activities might require interface configuration that is best supported through a full-featured desktop application. At this stage we are not interested in the design of desktop interfaces; only the mobile app. However, if features of your mobile app rely on configuration from a desktop interface (and cannot be configured through the mobile interface), then you should

make this clear in your design document."

What you should do

The key components of this assignment are as follows:

1. identification of representative users, including a discussion of the importance of supporting each user group;
2. identification of task categories, tasks within those categories, and specific uniquely enumerated scenarios for each task;
3. prioritisation of user tasks to be supported in the app, and associated rationale for those priorities;
4. sketched design alternatives, representing different ways of organising the app (note that each design is likely to be clearest if presented with the app's home page first);
5. design rationale for each of the approaches, identifying the pros and cons of each design;
6. an unambiguous specification of a final user interface design in wireframe form.

You are strongly advised to carefully read the "Interface Design" section of the course lectures, including "Task Centred System Design" (TCSD). Also, read the TCSD shareware book on Learn.

As stated in the lecture material, you should come up with as many early design concepts as possible (elaboration), and you should clarify and pursue the most promising ones (reduction). A small selection of the best sketched design concepts should feature in your report (around three alternatives would be wise).

Sketches will be very important in supporting and clarifying your intended design. Preliminary design alternatives should be legible hand-drawn sketches. Their purpose is to illustrate the intended interaction, so they should be sufficiently clear for the reader to understand how the main interface tasks are completed. It would be wise to scan (or photograph) your sketches, import them into a software application supporting vector graphics (e.g., PowerPoint), and embellish them with computer generated labels/callouts/arrows, etc. to identify important components.

The final interface design should be specified in wireframe format. It should *precisely* describe the design. I strongly recommend basing the design around the user's key tasks. The "Steve Jobs Patent" (available on Learn) provides an excellent example of how to annotate figures to make an unambiguous user interface specification:
https://learn.canterbury.ac.nz/mod/url/view.php?id=1244539

Don't get carried away with the patent style of language – you are writing a report, not a patent.

Work in teams of three to six

This assignment is most likely to be successful (i.e., you'll learn something) if it is completed in a team. I recommend teams of four or five. If you feel you have *strong* reasons for needing to work in a group of fewer than three or more than six, please

email me to seek approval before Sept 13th, and explain the reason for the request.

What you should submit

Exactly ONE member of each team should upload a PDF document of up to 40 sides to Learn – note, ONE submission per team. The submission must include the name and user codes of all team members, and the final page should contain a statement of the primary contributions of each of the team members.

Hints on document structure

Technical reports typically follow a similar format, including Title, Authors, Introduction, CONTENT, Conclusion. It would be surprising if your submission were to deviate substantially from this format. While there is lots of freedom over how to organise and present CONTENT, I would anticipate a structure similar to the following.

Title

Authors

Executive Summary/Abstract (*very* short summary of document purpose)

Introduction

User and Task Identification

> Including categories of users, categories of tasks, scenarios to exemplify tasks, and analysis of their priorities based on task frequency and importance.

Sketched Design Alternatives and their Rationale

> A few pages for each sketched design and its associated rationale, pros, and cons. Note that it's best if each design alternative shows a comprehensively different way of organising the user interface.

Final User Interface Specification

> A wireframe specification of UI, with accompanying rationale. It would be wise if the rationale explained how and why the UI supports easy-to-learn and/or efficient completion of the main user tasks.

Conclusion