

Handout for Example Classes 1 and 2

In Example Classes 1 and 2, you will design a **low fidelity (lo-fi) prototype** for an advanced task list application.

Scenario

Most of us would be familiar with various forms of task lists. Often these are simply written on paper; for example, a to-do list for the things that you need to do in school on a particular day, or a shopping list when you go to the supermarket. There are also various basic software that implement basic to-do lists (e.g. MS Outlook, Evernote, Asana, Trello).

However when planning a huge project, the corresponding tasks become much more complex. Such projects may include creating a large software system with multiple distributed programmers, or running a huge event like the *National Day Parade*, or organizing humanitarian activities in a foreign land.

There are various commercial project management software available today. They are often feature-bloated, with not only task list management but also integrated budget tracking and mechanisms for dealing with contingencies. Many of these are more heavily focused on providing features, rather than quality of the user interaction.

In this class, we will only concentrate on *user-friendly, user-centric designs* for an advanced user task list application, rather than a full project management system. The feature requirements of your task list application are described in a later section.

While your task list application should not be tailored only to specific project types, for this class you can demonstrate your application using the following scenario:

You are an employee in a company celebrating a special event. You are charged with planning and executing a company outing (say 100 people) to Gardens by the Bay that includes transport to/from the office, visiting the attractions and a catered dinner in one of the function spaces. For simplicity, you may ignore the budget.

Come up with a set of tasks that you think are needed to execute this scenario, and show how these can be planned and tracked using your proposed application.

*Your main focus for your design should **not** be on providing extra features, but rather on exploring different ways to enhance the quality of the user experience through nice interface visuals and intuitive user interaction. You should not assume you are limited in any way to software and platform constraints or other unnecessary conventions (e.g. ask yourself: do buttons have to be rectangular? Do they have to be arranged in a grid?)*

Task Item Specification

In your task list application, each task item should include the following attributes:

- Short title of task
- Description of task
- Start date
- Completion deadline

- Progress status (3 states: not started, in progress, completed)
- Task dependencies. This is a complex attribute that contains lists of other tasks on which this task depends. There are 4 categories:
 - Start-start. All the specified tasks must have *started* before this task can *start*.
 - End-start. All the specified tasks must have *ended* before this task can *start*.
 - Start-end: All the specified tasks must have *started* before this task can *end*.
 - End-end: All the specified tasks must have *ended* before this task can *end*.

Note that each task can have multiple subtasks (i.e. nested tasks) simply by manipulating the appropriate dependencies. For example, Task A can have a subtask Task B by: (1) specifying Task A as a start-start dependency of Task B, and (2) specifying Task B as an end-end dependency of Task A.

Key Features

In your design, you have to come up with design concepts for the following:

1. **Planning phase viewing.** Given an existing set of tasks (with dependencies), design ways for viewing the tasks. They can include viewing individual tasks in detail as well as a combined set in which it is easy for the user to quickly grasp how tasks depend on each other. They may include interactive mechanisms for viewing interesting different perspectives of the tasks and project.
2. **Planning phase modifications.** Design interactive ways for adding, deleting and modifying new tasks. Try and avoid mundane engineer-style ways of doing this, but rather ways that allow users to be intuitively aware how the new/modified tasks fit into the entire project, and how dependencies change or break with the modifications.
3. **Execution phase updates.** Design interactive ways in which the user can update progress on tasks, including starts and completions. The application should show which dependencies are triggered and also ways for the user to intuitively understand how the overall project is progressing, with delays and bottlenecks easy to notice.

How to Proceed

Here is what you should do in Example Classes 1-2:

- Do your preliminary preparation by reading the documents and watching the videos stated in the later section on *Preparatory Material*.
- Creatively think of the user interface most suitable for the features described earlier. You should take into account the following points:
 - The interface needs to enable the features as described above.
 - However, the description above is very general, and the examples given are neither complete nor mandatory. So that leaves a lot of freedom and scope for designing the user interface.
 - Interface design: what is the look-and-feel of the interface going to be? Does the interface follow the important principles of good interface design?
 - Experience design: what kind of experience will the user get out of interacting with the application? Are some forms of interaction more appealing and natural?

- *Ignore* all aspects of implementation feasibility for the time being: platform capabilities, programming language, compute power, memory, etc. Although these are eventually important practical issues, this exercise is to focus on thinking from the user's perspective – how to design for the best usability and user experience.
- Do *not* add excessive functionality features. Not only is feature bloat bad for usability, you are also *not* graded on number of features, but quality of the design.
- Sketch out one or more **low fidelity prototypes** of your design. Do not overly constraint your mind and your hand, just let different ideas come to you.
 - If you have multiple prototypes, you will (for practical reasons) have to select just one of the prototypes to submit for grading.
 - **Do not create a high fidelity prototype.** Not yet. For differences between lo-fi and hi-fi prototypes, please see the *Resources and Guidelines* section below.

Please note that this is **individual-based** work. Every student needs to finish the design and deliverables on his/her own. Discussions with other students or other people outside the class are allowed and in fact encouraged. However, make sure that the design appearing in your deliverables are **unique and completely developed by yourself alone**. **You should know that NTU very seriously emphasizes academic integrity.**

Deliverables

Every student must turn in **an individual submission**, with the following content:

- **Sketches of your lo-fi prototype design.** You can use **up to 4 A4 pages** to illustrate your design using a few sketches. Note here a “page” means one side of a piece of paper.
 - You need to submit both softcopies and hardcopies of your sketches. Please scan your sketches or at least take a *clear, high quality, high resolution* photo.
 - The sketches **must** be created by hand sketching (whether on a physical medium, or by stylus on a tablet and printed).

NOTE for those repeating this course: Unlike past semesters, the separate write-up has been eliminated, and the number of pages for sketching increased to 4. You are expected to make your design obvious based on your sketches alone, which can include substantial annotation.

Submission Instructions

Every student must submit their deliverables **seven days after the Example Class 2 session** (the recess week does **not** count). You must submit **both softcopies and hardcopies**. So by your **deadline**:

- You'll need to submit the physical copies to the **Hardware Projects Lab** (N4-01c-09a) during office hours, so safely before **5pm Mon-Thu** and **4.30pm Fri**, and then
- Upload the softcopies to NTUlearn by 11.59pm.
 - **IMPORTANT:** this is a hard deadline. Delays due to personal problems or technical faults (e.g. computer or network problems) is not a valid excuse for late submission. The only exception is if there is an NTU-wide network failure or NTUlearn downtime that started *before* 11pm of the deadline. **So please submit early.**

Do note:

- On your submission, especially for the hardcopies, you must first state **your own name** and your **CZ2004 lab group**.
- If you are **late** in turning in the deliverables, a coursework mark penalty will be imposed.
- You should make a **photocopy (or scanned copy)** of your design prior to handing in the documents. They will be used in session 3.

Assessment

This section provides some information on how the instructors will assess your submission.

Please note that the drawings handed-in **must** be created by hand sketching (whether on a physical medium, or by stylus on a tablet and printed), otherwise a heavy penalty will be imposed.

The assessment of the deliverables (i.e. your sketches) will be based on the following four components which are weighted equally:

- **Relevance of Design.** Does your design implement the specified features? You have the freedom to decide the form and manner of providing these features in your interface, so long as the intended functions of those features are kept.
- **Originality of Design.** Your design will be compared to the designs of other students in your cohort. The more unique your design is in terms of form, interaction and overall look-and-feel, the higher you will score for this component. Students with near-identical designs will likewise score near-zero for this component.
- **Quality of Design.** Does your design reasonably adhere to, or clearly violate, basic UI design principles, such as Schneiderman's Golden Rules? If you want to break a particular principle because you believe there is an exceptional reason, please consult your lab supervisor first to seek clearance, then clearly justify it in your submission.
- **Clarity of Submitted Material.** Based on your submitted sketches and report, is it reasonably easy for a reader to understand how the user will be interacting with your designed interface? Is there sufficient annotation to make the interface and interaction clear? Are important transitions diagrammed? Are there particularly outstanding illustrations of interaction and usage that deserve extra marks? Does the report clearly explain your design?

For lo-fi prototypes, don't worry about aesthetic details, such as fonts or color or background imagery. You will notice that aesthetics is **not** part of the assessment criteria. The assessment is based more on your interface following established design principles with interesting originality, good usability and providing an enjoyable interaction experience for users. Nevertheless, it is also important to have clarity in your sketches, otherwise the instructors won't understand your design.

Overall, you can expect that the effort you put into your design, your sketches and your report will be rewarded, and that greater effort will be recognized more clearly in these assessment components.

You should be aware from the course that lo-fi prototyping is a critical part of the design process because this is where very important choices of interaction and interface are explored before key decisions are made. This is in many ways more important than hi-fi prototyping. Hence please be aware that in the overall coursework assessment the lo-fi prototyping deliverables are given **greater weightage** than the hi-fi prototyping deliverables. So please allocate your effort and time accordingly!

Some future outlook: In Example Class 3, you will be critiquing someone else's work, and likewise you will receive a peer critique on your design. In Example Classes 4 and 5, you will design your hi-fi prototype. Your hi-fi prototype will need to be clearly developed from your lo-fi prototype, plus any critique comments that you think are appropriate. *You will not be able to create a hi-fi prototype that is not clearly linked to your lo-fi prototype design without losing substantial coursework marks.* So this is another point to note for taking your lo-fi prototype design seriously.

Resources and Guidelines

Here are some material, guidelines and suggestions that you may find helpful in your design work.

Preparatory Material

The concept of prototyping will be covered in our lecture, but perhaps not before you complete your lo-fi prototype. Hence it is strongly recommended that you do the following:

- Do your own reading up of the related materials in **module 3 of CZ2004** prior to embarking on the prototype design. This module has been uploaded to NTUlearn, and the related materials are in pages 12 to 23.
 - Please note that you do not need to be clear about all the concepts in these pages – they will be covered in detail in the lectures. You just need to understand: a) what is a prototype, which is obvious just by looking at the example prototypes provided in these pages; b) the difference between low-fidelity (lo-fi) and high-fidelity (hi-fi) prototypes so that you will **not** design a high-fidelity prototype at this stage. The difference between lo-fi and hi-fi prototypes is also quite easy to conceptualize from the examples (and descriptions) in the related materials.
 - The estimated time for reading this material is only about 10 minutes.
- Read the **Quick Primer on Sketching for Lo-Fi User Interface Prototypes** document on NTUlearn. This provides examples of lo-fi prototype sketches, and also suggests different things that you can sketch.
- Read and watch the **e-Learning material** provided on NTUlearn. These are very easy-to-digest material intended for laypersons. These include some inspiring videos that motivate and explain what interaction design and in particular lo-fi prototyping is all about.

What to present in your lo-fi prototype?

Please pay attention to the following aspects of your lo-fi prototype:

- **Key features.** Make sure that your prototype clearly illustrates the key features.
- **Annotations.** If certain features or functionality are not obvious from the drawings alone, please provide brief annotations to help viewers understand your design.
- **Temporal aspects.** When temporal considerations are involved in the desired features (e.g., a user action that can only be taken after another action), make sure that you capture in the sketch any temporal aspects of the interaction. That is, the sketch does not have to be like a “screenshot” of the user interface but rather it should depict the essence or spirit of the user interaction as much as possible.
- **Modes.** The student can also sketch different modes of the interface if it changes depending on the current state of the application. Say if you are designing a GPS navigator, the different modes may include a search mode and a navigation mode.

Refer to the **Quick Primer on Sketching for Lo-Fi User Interface Prototypes document** for further information.

How to prepare your lo-fi prototypes?

The most conventional way is to use pencil and paper, and draw your design on the paper. Please remember to **bring in your own paper and pen or pencil**. The electronics lab will not provide such materials.

An alternative is to draw electronically, say by sketching on a tablet with a stylus (do not use your finger as your fingertip is not fine enough!), e.g. Tablet PC or Galaxy Note, and then print the design on paper. Please note: (a) even if you choose to do this, you still need to turn in a hardcopy as well as an electronic softcopy; and (b) it is mandatory that you draw by hand sketching, not by using software tools.

It is also ultra-important to remember that we require lo-fi prototypes, **not hi-fi** prototypes. So, please pay due attention to the difference between a lo-fi and a hi-fi prototype.

Summary

Here is a summary of what you need to put in the individual report.

- Submit your lo-fi sketches, both **softcopies** and **hardcopies**.
- State your name and your CZ2004 lab group.
- Major content
 - **up to 4 A4 pages** of your *lo-fi* prototype sketches.
 - **Do not design a hi-fi prototype.**
 - The design must be prepared **solely by you**.
- Recommendation: self-study related materials beforehand.
- Turn the report to the Hardware Projects Lab on time (**1 week after** Example Class 2).
- Scan your paper sketches before you submit them.