

Sketching, Low-fi Prototyping & Pilot Usability Testing

Checkpoint by Tue, Oct 21 @ 5 PM

Final: Start of Thu/Fri studio (Oct 23-24)

Goal

Learn how to use low-fi prototyping techniques in the early stages of user interface design.

Assignment Overview

By the checkpoint (Tuesday 5pm):

- 1. Sketch out 20-30 solution ideas.** As a team, quickly explore a wide variety of modalities (e.g., speech, wearable, AR/VR, gestural, etc. – not just phone screens).
- 2. Of the 20-30 solution ideas, choose 3 distinct realizations.** The selected realizations should represent a diverse range of solutions across different modalities.
- 3. For each of the 3 selected realizations, create 3-5 concept sketches.** Create 3-5 rough sketches (e.g., screens) for *each* realization for a total of 10-15 sketches. See [this example](#), and aim for some novel realizations! The concept sketches should be well-annotated so that the intended user interactions are clear– tip: use color to help convey ideas!
- 4. Pick the top 2 diverse realizations and sketch 3-5 key screens for each.** The entire interface does not need to be fleshed out, but you should start to explore more of the user interface details than you might have in step #1. Specify the transitions between screens and add annotations to make the sketches understandable.
- 5. Pick your best idea with which to move forward.** To do this, make a list of at least 4 pros and 4 cons for *each* of the 2 realizations you fleshed out. These points should be based on your knowledge about your user base and design fundamentals from lecture and past assignments. Grounding your argument in the pros and cons list and the learnings from prior assignments, you should clearly explain why your final selected solution is superior to your other solution idea explored in part 2.
- 6. Storyboard 3 complete task flows: simple, moderate, and complex for your best idea.** These will show how each task is performed in your selected design realization. Label each screen, show the transitions, and include annotations explaining important interactions and states. Remember that you should define simple, moderate, and complex based on how difficult it is for the user to *accomplish* the task, not the difficulty of design/implementation. Note that the frequency of performing the task by the user can inform your goal for difficulty level (i.e., common tasks should be made simple for the user to complete!).
 - a. For each task flow, ensure that it is clear exactly how to complete the task; there should be enough screens and detail to communicate how the UI would work.
- 7. Construct your low-fidelity paper prototype.** Use the techniques described in the [Snyder chapter](#) (pwd: hcid) as a guideline..

- a. There should be a sufficient number of screens to construct 3 complete task flows. Complex graphics (vs. sketches) should only be used when absolutely necessary, but do include widgets and overlays (drop down menus; text fields; dialogs) to facilitate a realistic experience of interacting with the UI. For example, if a prototype requires scrolling/swiping motions, the paper prototype screens should be able to support responsiveness to these motions.

- 8. Describe your low-fidelity paper prototype.** Discuss what was used to construct the prototype (e.g., Paper vs cardboard) and how the prototype operates (i.e., team member roles, environment, etc). Pictures always help with descriptions. Explain what decisions were made during low-fi prototype construction, and why those decisions were made.
- 9. Submit your low-fi prototype via Slack to your CA by the checkpoint (Tuesday, 5 PM) for feedback before you test.** This checkpoint helps ensure that your prototype is of the right fidelity so your testing won't be in vain and is of the appropriate level of novelty. We encourage you to submit your low-fi prototype to your CA as soon as possible (yes, you can submit your prototype *earlier* than the checkpoint due date!). The earlier you submit to your CA, the sooner you will be able to start testing. While you wait, you should finalize scheduling your testing participants and work on your presentation deck.

By the final due date:

- 10. Implement any changes and test your prototype with at least 1 participant per team member.** For clarity, if you have 4 team members, you are responsible for 4 tests, etc. Participants should fall within your target user base. You may test with **at most 1 Stanford student** (preferably none) unless you have permission to do otherwise from your CA or Professor Landay. Interviewing a Stanford student may limit your ability to achieve an above-average grade for this part of the assignment. Avoid testing with people you know well and people who have already seen your project. Have each participant sign a [consent form](#) (copy and edit this document).
 - a. *Participants:* Explain how each participant falls within the intended user base, how they were recruited, any compensation given. Include images of participants interacting with the prototype while the tests were being conducted.
 - b. *Environment & Apparatus:* Explain where the test was conducted and which technologies/equipment were used. Include images of the set-up.

11. Testing Procedures

Pre-testing Preparation

- **Determine team member roles.** Include a thorough description of the procedure with images of the test being conducted. Team member roles and descriptions should be clearly outlined in this section. For example, facilitator, observer/note-taker, or computer. Practice these roles together in advance so everything runs smoothly.

- **Set up/plan your testing environment.** If remote (only if pre-approved by your CA), send your participants a link to your prototype and use one of the following methods to observe the testing session:
 - Participant runs the prototype on their device and screen shares. Note: if your prototype is for a mobile app, try to get participants to test on a mobile device, etc.
 - Participant joins Zoom and “hugs” their laptop so you can see their phone screen. See this [article](#) and refer to lecture slides for further explanation of the techniques.
- **Write a script for your demo and test procedure.** You should follow the same script with each participant to ensure validity across your tests.
- **Devise goals and metrics to assess the effectiveness of your prototype.** Come up with **2 usability goals** (see [Lecture 1](#) and parts of [this reading](#)) and a **key measurement** for each that will help you continually assess whether your prototype is achieving its goals. Include both *process data* (i.e., what is happening) and *bottom-line data* (e.g., time spent per task or # of errors). These will be revisited in A6 as you iterate. Explain why the selected usability goals are logical based on the current progress of the project, and what these goals mean for your particular solution moving forward.

In-test Procedure

- **Briefly demo the system.** At a high-level, show the participant how they can interact with your prototype (e.g., how buttons or scrolling works), but do not show them how to perform any tasks. *Demonstrate speaking aloud.*
- **Have your participant test your tasks.** Tell them what they are trying to achieve, **not how** and provide adequate context. Only give them task directions for a single task and test one task at a time and remind your participant to think aloud. Make sure your observer is taking note of both positive and negative critical incidents that occur (e.g., mistakes, quotes, emotions, etc.) . When the participant finishes a task, give them directions for the next and so on.
- **Wrap up.** Collect any final thoughts and answer any questions they might have.

Post-test Procedure

- **Synthesize a log of critical incidents.** These will include both positive and negative events during the test. Prioritize these incidents by assigning each a severity rating (0 = no problem, 1 = cosmetic problem, 2 = minor usability problem, 3 = major usability problem, 4 = usability catastrophe). Discuss the implication of these findings for the project moving forward.
 - Justify severity ratings using the process and bottom-line data from the pre-test preparation. Add evidence to back up findings (e.g., images or quotes) and use graphics/charts to represent the data.

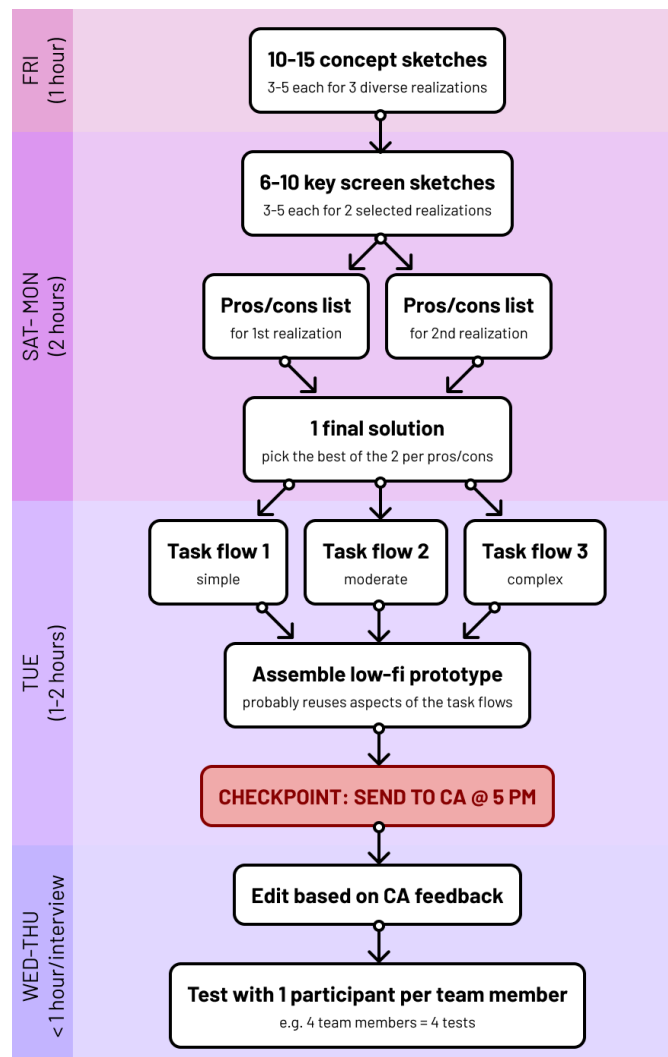
- Consider how the results vary among different users, and reflect on the validity/applicability of the results.

12. Discuss Low-fi Testing Results

- Based on the results of low-fi testing, map how the findings will impact the UI design moving forward. Justify your design changes, and feel free to visually display how you would change the design (e.g., a rough ‘before’ & ‘after’ sketch showing the changes to be made to your design based on the testing results).
- Discuss progress toward the outlined usability goals with respect to the key measurements (from part 9).
- Acknowledge any shortcomings of the low-fi testing methodology. Think about what the experiment revealed versus didn’t reveal and how that impacts test validity. What does this mean for your solution moving forward?

Suggested Timeline

This assignment is dense! Start as soon as possible to complete all steps in a manageable timeframe. Below is a recommended timeline.



Sketching Report Guidelines

Since it's difficult to see the details of a sketch when included as an image in a slide deck, we ask that you produce an additional document. All images should be large enough to see the details and captioned and annotated as required for understanding by an outside audience (your CA).

Expected Content

1. Title page
 - a. Project title
 - b. Value proposition
 - c. Team member names
2. Concept sketches
3. 2 further fleshed out realizations
4. Low-fi prototype
 - a. Birds-eye view image of the entire system
5. Task flows with transitions and interactions labeled

Presentation Guidelines

One team member who has not yet presented will present in studio. There will be **11 minutes** for the presentation and 5 minutes for questions and feedback.

Expected Content

1. Intro
 - a. Project title and team
 - b. Value proposition
 - c. Problem/solution overview
 - d. Outline of talk
2. Sketching explorations
 - a. Overview images of 10-15 concept sketches – *no need to talk in detail about these while presenting, just meant to capture your process*
 - b. 2 further fleshed out realizations
3. Selected interface & rationale
 - a. Present the most relevant pros and cons and have the full lists in the appendix
 - b. Relevant data, constraints of the platform, and findings that shaped your decision
 - c. Why is the design you chose superior?
4. Low-fi prototype construction
 - a. How the prototype was built and operated; its features and interactions
5. Low-fi prototype: 3 task flows
 - a. Simple, moderate, and complex
 - b. Complete and clear with use of captions, labels, transitions, and annotations as required (can use a different color if helpful)
6. Testing methodology
 - a. Participants: demographics, recruitment, compensation

- b. Environment and apparatus
 - c. Procedure: team member roles, description of process
 - d. Usability goals and key measurements
7. Testing results
- a. Process data (what's happening in the big picture)
 - b. Bottom-line data (usability goal key measurements)
 - c. Other relevant observations
 - d. How well did you achieve your usability goals based on the key measurements?
8. Discussion
- a. What are the implications of your findings?
 - b. Based on the results, how will you change your design?
 - c. Was there anything that the testing couldn't reveal?
9. Appendix
- a. Full list of pros and cons for selected interface rationale
 - b. Link to the low-fi prototype on Marvel POP, etc. (if applicable)
 - c. Any preparation you did for the testing (e.g., script)
 - d. Organized log of critical incidents from testing
 - e. Other figures, etc. that might be helpful for your CA

Deliverables

Upload deliverables to a subdirectory titled "Assignment 5" in your team's Google Drive folder.

1. Presentation

Google Slides deck in PPTX and PDF format presented by 1 team member during studio.

2. Participant consent forms

Combined into a single PDF.

3. Sketching report

Google Doc in PDF format with the required information outlined in this spec.

Examples

***Note: this assignment has been modified, so these examples are not perfect mappings to the deliverables; however, much of the quality of the work stands.*

[Turno](#), [playdate](#), [Budder](#), [StoreaTime](#), [SleepMate](#), [Journy](#), [QuokkaTips](#), [Voices](#)

Grading Criteria

Grading is broken into 2 components: 1) a group grade for the slides and sketching report content and 2) an individual grade for the presenter and their slide quality.

Group deliverables (100 pts)

Checkpoint (5)

- Low-fi submitted for feedback on time and is relatively complete
- Feedback was acted on appropriately

Sketching (20)

- 10-15 concept sketches that capture 3 diverse realizations
- 2 realizations are fleshed out; key screens are understandable

Final design rationale (5)

- Pros/cons lists for 2 selected realizations
- Final rationale is logical and clearly explained

Task flows (15)

- One each of simple, moderate, and complex
- Quality, low-fi task flows with necessary screens/details

Low-fi prototype (20)

- Complete and built using low-fi techniques
- Clearly explains how the prototype was built and its features
- Reflects exploration of novel interface design

Usability testing method (20)

- At least 1 usability tests/team member were conducted
- Participants: demographics, recruitment, and compensation
- Testing environment and apparatus are appropriate, images included
- Testing procedure is clear and appropriate, images included as available
- Usability goals and key measurements are logical and thoughtful

Results and discussion (15)

- Relevant data, critical incidents, and images are included
- Reasonable UI changes are recommended based on testing results
- Discusses progress made towards usability goals
- Shortcomings of the usability test; what might it have failed to reveal?

Presentation grade (100 pts)

- Well-designed slides; visual aids are aesthetic and effective
- Covers required scope within 11 minutes
- Engages with the audience and isn't reading from a script
- Projects voice well and communicates clearly