



*Paper prototypes for three versions of a home thermostat. From [Tohidi CHI2006].*

## Important notice — Read carefully

This week, you will start working on what will become your Web site. Remember, this class focuses on interaction, not implementation. We encourage you to make the coolest application you can, but don't require textual programming. If you're most comfortable with graphical tools, we recommend Webflow. (You're of course welcome to use text tools if you prefer; the choice is yours.)

Please note that you will not start using a high-fidelity tool until the next assignment (right now, you are just prototyping in Balsamiq or an equivalent low-fidelity tool); this is just about thinking for the future. Consider what interactions are most important to your application and what prototyping tools support those interactions. Think about that now so you don't get surprised when designing later.

Here's the strategy for this assignment

1. Pick a concrete direction and
2. Create 2 rapid interactive prototypes
3. Finally, the peer evaluation will introduce you to the concept of **Heuristic Evaluation**.

# 1: Choose a Storyboard

Decision time. Set out your storyboards. Take a **deep breath**. Spend some time to **reflect on** the different ideas you've had. Make sure you discuss the strengths and weaknesses of each design, and how well they achieve the goals set out by your point of view. And then, **decide on one** of these ideas to prototype.

# 2: Create 2 Rapid, Interactive Prototypes

Make **two rapid interactive prototypes** that **implement the idea** you've decided on. (Can you see we really value enumerating alternatives?) A rapid interactive prototype concretely **shows all the major elements** of a user interface, except that it's implemented with very high level tools (just like pen and paper), as opposed to actual code. This is often called a 'Wire Frame', or '**Wireframing**.'

Rapid interactive prototypes can be done with [Balsamiq Mockups](#). If you're used to other prototyping tools, feel free to use it as long as it's a **low-fidelity prototyping tool**! (You may want to install Balsamiq anyway to see what kind of prototype you want to make for this assignment.)

**No Photoshop, no Illustrator, no polished work, even if you are familiar with them!**

Again, it helps **focus on the concepts**, and saves you from **wasting hours twiddling pixels**. If you haven't understood that yet, the goal is to **be fast**. Years after taking this course at Stanford, students often come back and tell us that rapid prototyping (whether on paper or electronic version) was their favorite part of the class because of **its effectiveness for rapid ideation**.

The prototypes you create **should be interactive**. That means that your user should be **able to navigate** around it and **get a feel** for how the 'real thing' would work. This is not as complex as it sounds!

Balsamiq's built-in tools allow you to create "links" between screens and buttons to simulate real functionality - back, next, home, etc - even though visually it is nothing more than a line drawing.

The prototypes should **vary in interface**, but offer the **same basic functionality**. For example, if you were designing a mobile transit application, your two prototypes could display the bus times in two very different ways. The prototypes should be **complete enough** to "run" a new user through each task. (Note: the **story here is very linear**, but your **process doesn't have to be**. You can start making a rapid prototype, then change your mind. And your prototype **doesn't have to exactly align** with your storyboard.) Your prototype interface should enable people to **navigate, recover from errors, and change** their mind.

Remember that your peers will be grading you on **completeness of your prototype**, so you'll be **marked off** for any functions that you've included that aren't interactive. Try to show **all important areas** of your site that relate to the major information architecture and functionality. If X is an area of your site that's not very relevant to your task (like maybe the copyright policy isn't very important for designing the use case of this class) then you don't need to show it.

*(Although we are not going to do any paper prototyping in this assignment, being familiar with this concept is interesting. For further information, see the book [Paper Prototyping](#) by Carolyn Snyder and a few [references](#)).*

## **Student Examples**

- These two prototypes show two different interfaces for creating and tracking goals (while also experimenting with full pages vs. light boxes for certain entry forms): [1st prototype \(18-4fd7ae7f6568b9.96995878.pdf\)](#) and [2nd prototype \(18-4fd7bf94e602f7.45478747.pdf\)](#)

## ***What's this for? A UX agency perspective***

*by Mike Davison, Community TA and UX Project Manager*

*Wireframing is an important step in the design process. You may know how you want your application to look and wish to skip ahead, otherwise, you need to do the work twice, right? But **this step is vital**. It allows you to decide on navigation, layout, menu options, application structure and much more, making quick changes as you go.*

*Importantly, it allows you to **TEST your plan** with users **before you spend many hours** on visual design. I recall a recent project that sprung some major surprises here; the user base disagreed fundamentally with the ‘needs’ their corporate management had identified for them. It prompted a **drastic redesign**...if the visual layer had been applied, we would have **wasted tens of thousands more**.*

*Think about it as if you were **building a car**. You do not **start painting** body panels and trimming the interior **before** you decide **where the engine** is going to go!*

## **Submit**

Your write-up will contain:

- Digital photos or scans of the **storyboard** you chose.
- Your **2 rapid interactive prototypes** (if you used Balsamiq, this will be an interactive PDF you upload).

# Common "I wish" Feedback

The following statements are common feedback given on this assignment. We call these statements "I wish" feedback because they are a way to express things that you wish the assignment had contained. You could think of these as **common problems to avoid**.

- Prototype idea should try to **address user needs** in ways that are **not similar** to features of commonly used services.
- Prototype(s) should **relate to the user needs** in the storyboard more.
- Prototype(s) should have **clearer interactions**.
- Prototypes should **diverge** more.



# Evaluation criteria & Grading rubric

The peer evaluation will introduce you to **Heuristic Evaluations** (HE). As you are grading your peers, we will ask you to conduct heuristic evaluations (HEs) of the prototypes in order to complete the prototyping phase of the project, providing you with the feedback you need to begin implementing. [Check out our discussion of Heuristic Evaluations in our lecture videos \(4.1\).](#)

## Heuristic Evaluations

The heuristic evaluations will be a way to **highlight usability issues** in your rapid interactive prototypes. Heuristic evaluations will follow the ["How to Conduct a Heuristic Evaluation"](#) and ["Ten Usability Heuristics"](#) readings by Jakob Nielsen.

Using **Nielsen's ten heuristics**, you as an evaluator will list as many usability issues as possible. It is less important that the evaluators worry about exactly which heuristic the usability issues occur in. The heuristics are there to help evaluators **find a variety of different types** of usability issues. For each usability issue found the evaluator should **capture a description of the issue** written as **concisely** and as **specifically** as possible.



## Being an Expert Evaluator

As an evaluator, **be thorough** and write down all problems you can find. **Don't try to be "nice"** by not reporting problems, everything you find will **help your peer improve** his interface. Since you are evaluating two prototypes, try to make your **feedback comparative**. For example, if a prototype had a big problem that you also found in the other prototype, make a note of it. If one prototype had a problem that, for some interesting reason or another, was solved or not an issue in the other prototype, make a note of that as well. You don't have to compare both prototypes in every sentence in your evaluation (and it's even fine if most of your feedback isn't comparative), but you should **highlight enough similarities and differences** between the two prototypes so that the peer receiving your feedback will understand the advantages and drawbacks of each design.

Your feedback should **help your peer understand** and **decide** which features of which designs they should implement in the coming weeks. Use Nielsen's heuristics as a guide and specify what heuristic(s) each problem is related to (but if you come across problems that don't fall neatly into particular heuristics, **note them down also!**). In addition, write the problem down with a severity rating as in Nielsen's [Severity Ratings for Usability Problems](#).

**Go over both of your peer's prototypes.** When grading your peers, write a heuristic evaluation of the two prototypes. This comprises a bulleted list of usability issues you found, along with their severity, for each prototype. Include comparative feedback between the two prototypes.

## Evaluation criteria & Grading rubric

Category	Unsatisfactory	Bare minimum	Satisfactory effort & performance	Above & Beyond
<b>Prototype #1: Clarity</b>	0: No prototype or completely irrelevant prototype.	1: Many elements have no defined purpose, and it can be difficult to know how to use many parts of the prototype.	3: Some elements have no defined purpose, and it can be difficult to know how to use certain parts of the prototype.	5: All elements have a clearly defined purpose and it is easy to know how to use the prototype. A programmer could use the prototype to create a functional application with a defined flow.
<b>Prototype #1: Completeness</b>	0: No prototype or completely irrelevant prototype.	1: The prototype is incomplete, as it is not interactive or interactions are broken.	3: The prototype is somewhat complete, although there are some functions that are not yet interactive.	5: The prototype is complete, with all relevant functions interactive.
<b>Prototype #2: Clarity</b>	0: No prototype or completely irrelevant prototype.	1: Many elements have no defined	3: Some elements have no defined purpose, and it	5: All elements have a clearly defined

		purpose, and it can be difficult to know how to use many parts of the prototype.	can be difficult to know how to use certain parts of the prototype.	purpose and it is easy to know how to use the prototype. A programmer could use the prototype to create a functional application with a defined flow.
<b>Prototype #2: Completeness</b>	0: No prototype or completely irrelevant prototype.	1: The prototype is incomplete, as it is not interactive or interactions are broken.	3: The prototype is somewhat complete, although there are some functions that are not yet interactive.	5: The prototype is complete, with all relevant functions interactive.
<b>Prototype divergence</b>	0: No prototypes to compare.	1: The prototypes do not vary in interface in any significant way.	3: The prototypes somewhat vary in interface, or they vary in interface but address different ideas.	5: The prototypes explore clearly different interfaces which address the same idea.

**Total: 25 points**

To give your peers more context when evaluating you, please explain the activity and design brief you chose and how they relate to each other.

Write down your Point of View.

Upload the storyboard you chose in Step 1.

Upload (or link to) an interactive PDF of your first prototype.

## Evaluation/feedback on the above work

**Note:** this section can only be filled out during the evaluation phase.

Prototype 1: Clarity

- **0 points:** No prototype or completely irrelevant prototype.
- **1 point:** Many elements have no defined purpose, and it can be difficult to know how to use many parts of the prototype.
- **3 points:** Some elements have no defined purpose, and it can be difficult to know how to use certain parts of the prototype.
- **5 points:** All elements have a clearly defined purpose and it is easy to know how to use the prototype. A programmer could use the prototype to create a functional application with a defined flow.

Brief feedback:

Prototype 1: Completeness (Note to evaluators: If the prototype does not appear interactive when viewing in browser, please download the PDF and test it. Some browsers do not appear to support the interactivity.

- **0 points:** No prototype or completely irrelevant prototype.
- **1 point:** The prototype incomplete, as it is not interactive or interactions are broken.

- **3 points:** The prototype is somewhat complete, although there are some functions that are not yet interactive.
- **5 points:** The prototype is complete, with all relevant functions interactive.

Brief feedback:

Upload (or link to) an interactive PDF of your second prototype.

### Evaluation/feedback on the above work

**Note:** this section can only be filled out during the evaluation phase.

Prototype 2: Clarity

- **0 points:** No prototype or completely irrelevant prototype.
- **1 point:** Many elements have no defined purpose, and it can be difficult to know how to use many parts of the prototype.
- **3 points:** Some elements have no defined purpose, and it can be difficult to know how to use certain parts of the prototype.
- **5 points:** All elements have a clearly defined purpose and it is easy to know how to use the prototype. A programmer could use the prototype to create a functional application with a defined flow.

Brief feedback:

Prototype 2: Completeness (Note to evaluators: If the prototype does not appear interactive when viewing in browser, please download the PDF and test it. Some browsers do not appear to support the interactivity.

- **0 points:** No prototype or completely irrelevant prototype.
- **1 point:** The prototype incomplete, as it is not interactive or interactions are broken.

- **3 points:** The prototype is somewhat complete, although there are some functions that are not yet interactive.
- **5 points:** The prototype is complete, with all relevant functions interactive.

Brief feedback:

Prototype divergence

- **0 points:** No prototypes to compare.
- **1 point:** The prototypes do not vary in interface in any significant way.
- **3 points:** The prototypes somewhat vary in interface, or they vary in interface but address different ideas.
- **5 points:** The prototypes explore clearly different interfaces which address the same idea.

Brief feedback:

## Heuristic Evaluation

Write a heuristic evaluation of the two prototypes. This comprises a bulleted list of usability issues you found, along with their severity, for each prototype. Include comparative feedback between the two prototypes.

For each issue you find, you should include:

- Which [heuristic](#) has been broken
- Where you saw the problem
- A description of what the problem is
- A [severity rating](#)

[Click here for an example of heuristic evaluation feedback \(ExampleAssignment3HE.pdf\)](#) a student received. Try to aim for the type of feedback the student gave herself and that given by Student 3.



## Overall evaluation/feedback

**Note:** this section can only be filled out during the evaluation phase.

### "I wish" Feedback:

How could this student best improve his/her prototypes? From among the following, copy one or more pieces of advice that would help the student. Paste your advice in the feedback box below.

- Prototype idea should try to address user needs in ways that are not similar to features of commonly used services.
- Prototype(s) should relate to the user needs in the storyboard more.
- Prototype(s) should have clearer interactions.
- Prototypes should diverge more.
- Other

Optionally, for each item you choose, say why this feedback will make the prototypes better.