



DESIGN PORTFOLIO

ROB FAVRE

Low fidelity prototypes 1

High fidelity prototypes 3

Documentation for controls and components 5

Feature documentation 6

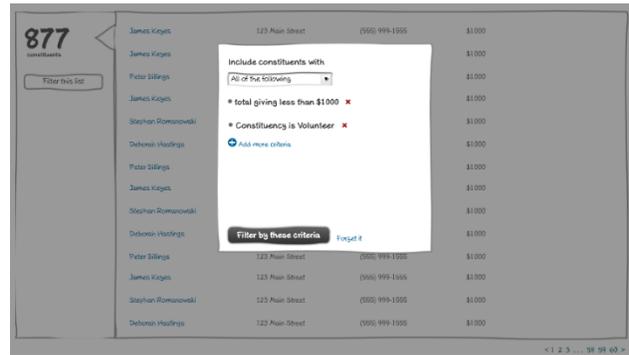


Low fidelity prototypes for learning

Early in the design process, it's valuable to generate and iterate on ideas quickly. Low-fidelity prototypes are a good tool for this, since they don't require a lot of time spent on visual polish. They have other benefits as well. Here are some examples from projects I've worked on.

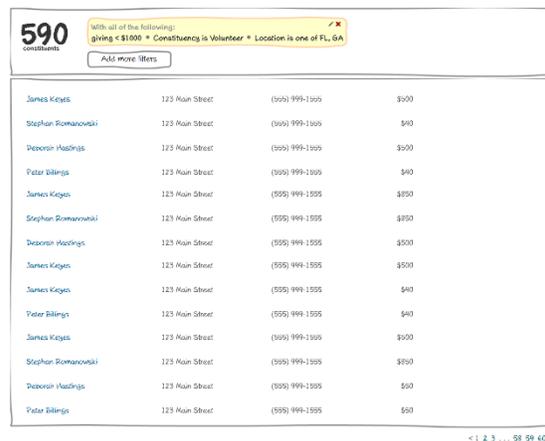
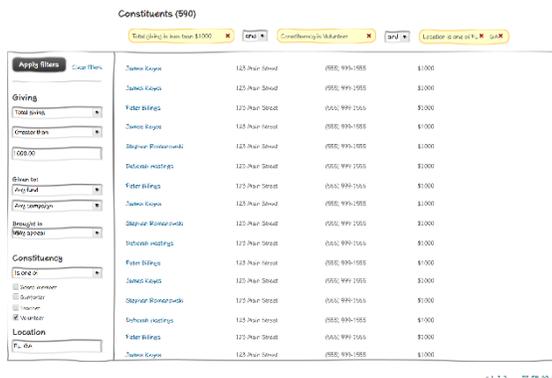
List filter prototype

Axure has a nice feature that I like to use for certain projects that will let you make your prototype look "sketchy" while keeping all the interactivity of a high-fidelity prototype. This isn't the best way to generate ideas quickly, but it does help reassure users that they're looking at an early concept and not a "real product," which sometimes generates more honest feedback.



This project was about applying filters to a list, and then having awareness of which filters had been applied.

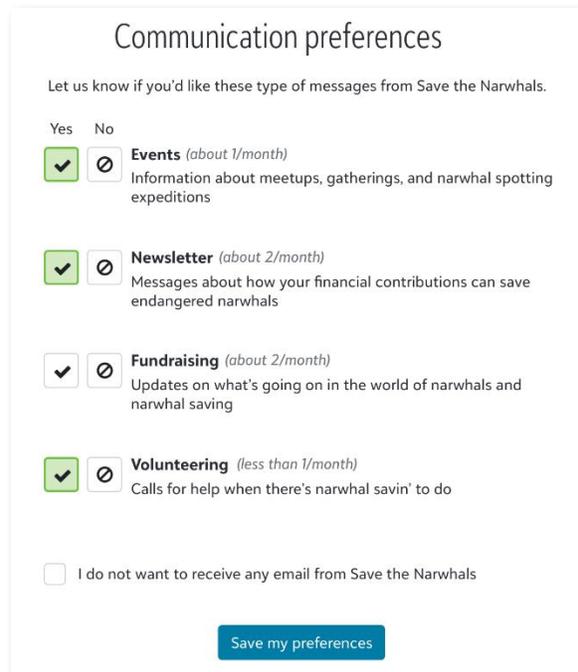
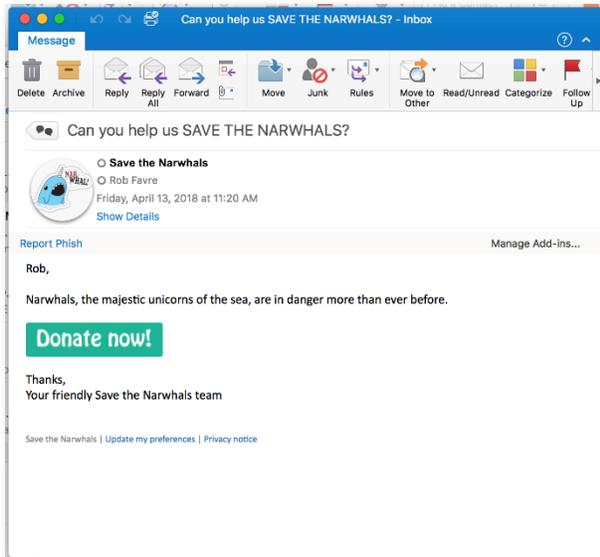
Clicking each screen will open the corresponding interactive version. The scenario the prototypes are meant to test is to apply filters by Giving, Constituency, and Location.



Email opt out concept

A simple prototype to test a workflow for email recipients to set their communication preferences. We put it in front of some communications directors to see if they liked the way the options were presented to the user. We also guerilla tested it with coworkers around the office to get a sense for whether the options made sense as presented.

This was built using Sketch and Invision. Click on one of the images below to view the full prototype.



High fidelity prototypes for usability testing

For validating the usability of a near-finished design, it's best to create a prototype that looks and behaves exactly like the finished product will.

Family data entry

Some of our church customers faced a problem: it was taking a long time for the Sunday school staff to enter data about new families dropping their kids off for the first time. Because the system's child safety features rely on accurate data, it was crucial that staff with minimal training be able to quickly and correctly enter important data about each family. I worked on a design to streamline that entry process.

This prototype was built to test the part of the process that creates relationships between the family members. It features built-in instructions about the scenario to ensure it could be used with minimal facilitation.

Add family

Address
460 Hickory Street Ashtabula, OH 44004

Family relationships

- Parents to children
- Parents to each other
- Children to each other

Parental relationships are the same for each child It's complicated

Parents

Molly is the of all the children

Frank is the of all the children

Children

Hunter is the of both parents

Rebecca is the of both parents

Family summary

Molly Seymour	(831) 363-6054
Primary contact	molly@notmail.com
Frank Seymour	(831) 363-7809
Spouse	
Hunter Seymour	
Child	
Date of birth	5/21/2009
Grade	5th
Class	5th grade B
Rebecca Seymour	
Child	
Date of birth	3/1/2013
Grade	1st
Class	1st and 2nd

Remember:

- Molly and Frank are married
- Hunter and Rebecca are their children

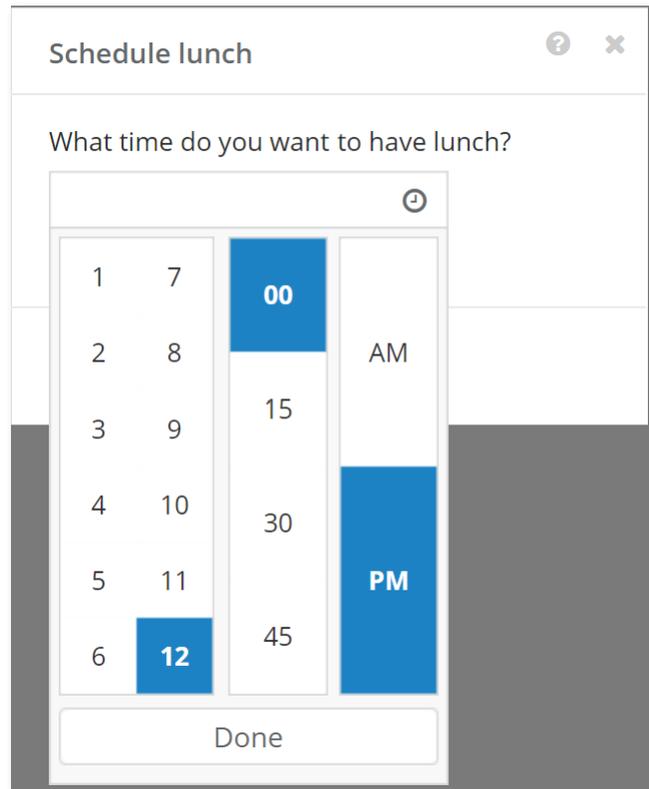
Save Cancel

Time picker control

This was a component that I designed for inclusion in the [SKYUX open-source component library](#). We wanted to have a time select control that functioned just as simply and naturally as our date select control, and was easy to use on mobile/touch devices.

The prototype, which was designed to be shown on a mobile device, includes three different methods for selecting a time. Users were instructed to select a time with each and when all three were completed, the moderator could click the “Show results” button to display how long each selection took.

Click the image to see the prototype in action. You can see the [finished version of the component on the Blackbaud developer page](#).



Documentation for controls and components

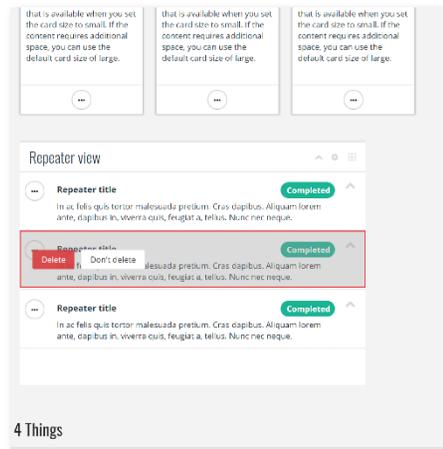
When I create documentation about controls and components, it really serves as content for two different audiences. My fellow UX designers want information about the purpose of the component in their designs and guidelines for its use; the developers tasked with building the control want details about the interactions and visual styling.

Here are couple examples of component documentation I built in Axure that use a hybrid approach: an interactive demo to show behavior and more traditional documentation to explain how to style the control and when to use it.

Inline delete

This component confirms deletion of an object with a lightweight inline message that keeps the user in context and doesn't require a disorienting full-screen modal message.

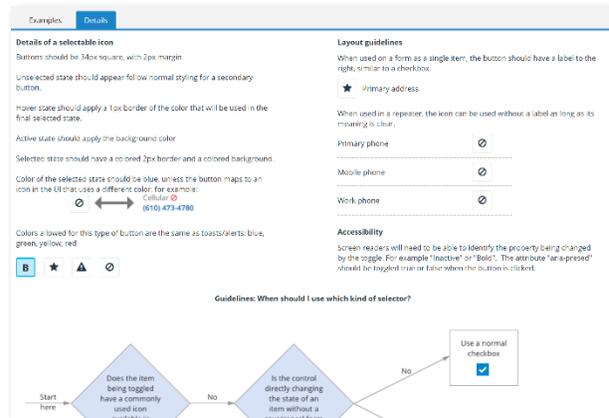
Click the screen to see the full prototype, or you can see the [finished version on the Blackbaud developer page](#).



Selectable icon buttons

This was a pattern document that standardized a control several teams had implemented independently. Creating a common, reusable component allowed for predictable behavior for end users and less waste and overhead for the engineering teams.

Click the screen to see the full prototype, or you can see how it's been incorporated into several standard components on the Blackbaud developer page: [radio buttons](#) and [checkboxes](#).



Feature documentation

Interactive prototypes are invaluable for sharing design ideas with stakeholders and testing them with users, but they leave much to be desired as a spec for an engineering team to follow. Specific styles, responsive behaviors, error cases and edge cases are all sometimes better handled by just calling them out explicitly. This kind of documentation is very helpful for a team to have available before and during sizing exercises so they aren't surprised by "hidden" requirements popping up during the course of a sprint.

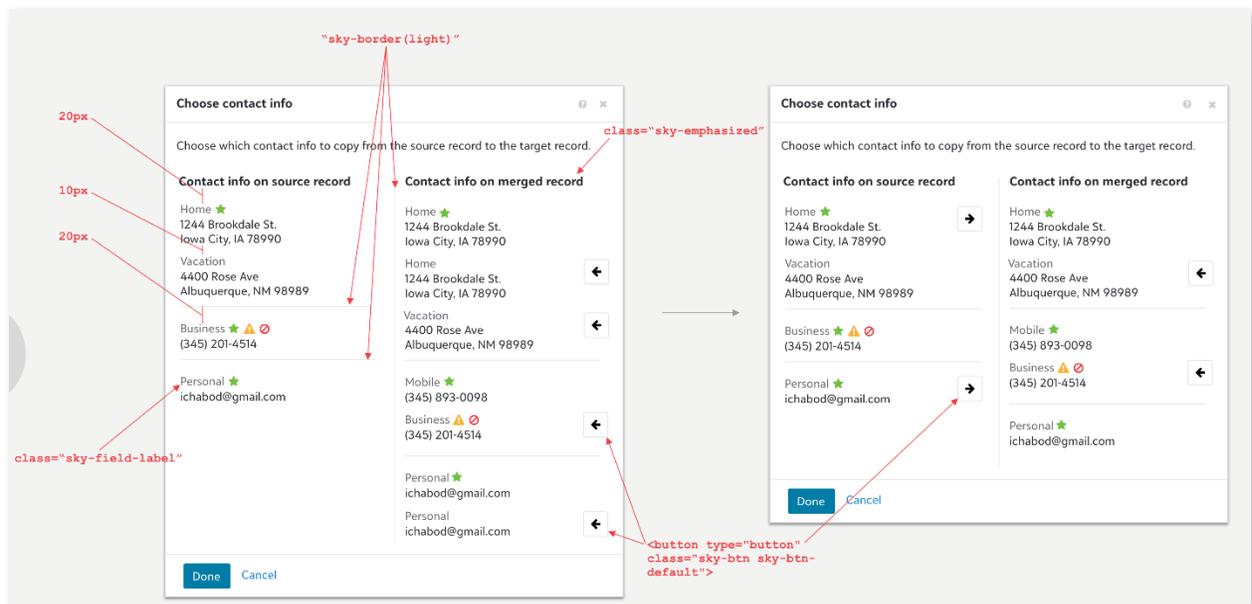
Some of these mockups were nominated for a peer award by an engineering team working with me, who said "His efforts to create exceptional mockups are a breath of fresh air and should set the example for how designs should be done in the future."

Record merge documentation

I worked for quite a while on a feature that let a system administrator detect when two records were actually duplicates representing a single person and then merge them together. Part of that process involved picking through specific data on each record to decide which pieces of each were worth keeping on the final, merged record.

This is documentation for the part of the UI that lets the user merge contact information from the two records. It includes styling details and calls out how to handle specific detailed cases.

Built using Sketch and Invision. Click the image to view the full page.



Styling update documentation

As part of a product-wide style refresh, our engineering teams had to replace a lot of old components with newer versions. Sometimes the old and new versions didn't exactly match, and sometimes there was no obvious match in the new library for existing functionality. I went through many of these screens with a fine-toothed comb and called out exactly which new components should be used to replace the old ones and how the new ones should be used.

Built using Sketch and Invision. Click to see the [full version](#).

edit addresses.

