The Information Architecture Behind Good Web Forms

by Luke Wroblewski

Information Architecture

During a recent interview, I was asked to distill the hundreds of web form design best practices I had been researching and documenting into my "top three tips for designing good forms." It wasn't until a few days later that I began to realize what all three of my recommendations had in common. They were all information architecture challenges.

While others may see things differently, for me information architecture challenges require extensive thinking about the structure and flow of information, which include information flow into and out of software applications. So it's not hard to see how forms fit into the mix and why I honed in on the relationship of information architecture to these tips.

Tip 1: Design the Questions You Are Asking

When designing or developing web forms, people are often quick to take the questions within an existing paper form, desktop software solution or requirements document and simply translate them into labels and input fields on a web page. This translation is how most web forms are "designed." Much less common is the process of taking a hard look at the questions a form is asking and pruning what is not required or can be deferred until later.

Every question asked by a form requires someone to parse it, formulate an answer and then enter their answer into the affordance provided by an input field. Anytime you can remove or defer a question, this process is avoided and people can make their way through forms faster and with less effort. As a result, judiciously culling the questions being asked can reduce form completion times and increase completion rates.

Luke Wroblewski (LukeW) is an internationally recognized web thought leader who has designed or contributed to software used by more than 500 million people. He is currently senior director of Product Ideation and Design at Yahoo! Inc.

Designing the questions you are asking in web forms is an information architecture challenge because it requires us to think about the data set that will allow people to accomplish their goals. No one really wants to fill in a form; they want what is on the other side: to buy a book, to register to vote, to join a club. What set of information allows them to accomplish this goal with minimal overhead and a clear sense of purpose so they don't stop and think, "Why am I being asked this?"

Analyzing people's goals in order to understand what information they require to achieve those goals, then structuring that information into a logical series of questions is clearly different from picking the labels and input fields needed to represent a pre-existing set of questions. This crucial difference can amount to a big impact on form completion metrics.

Tip 2: Design a Clear Path to Completion

It is one thing to understand what set of information will get people to their goals; it is another to organize it in a way that illuminates a clear path to completion. Because people want what's on the other side of a web form, their general tendency is to jump right in, start answering questions and hope to get done quickly.

As a result, forms need to be organized in a way that makes it abundantly clear how they can be completed. At first glance, people should be able to understand what they need to do and how. Forms that scatter questions around the page and don't provide a logical ordering between information requests impede people's ability to quickly formulate answers as they progress through a form.

Messaging without proper priority, like hard-to-find error messages, and unconnected primary actions can similarly cloud the steps people need to take in order to get through a form. These issues and several more are

WROBLEWSKI, continued

FIGURE 1. A product replacement form by Nintendo Inc.

present in the form by Nintendo shown in Figure 1, which features a mix of random visual emphasis, poor affordances and a form sandwiched between excessive legalese.

Illuminating a clear path to completion is an information architecture challenge because it requires us to structure information requests in a way that makes it abundantly clear how they can be completed; in other words, we must tee up questions in a logical pattern from start to finish.

Tip 3: Design Engagement, Not Forms

Getting people through a form is just one way to gather information. On the web, however, forms are often the de facto choice for data requests. But it doesn't have to be this way. Instead of throwing a web form in front of people every time we need some information, we can turn instead to the principles behind gradual engagement.

Gradual engagement allows us to gather information from people in a way that gets them involved in our web applications. If done right, gradual engagement can also educate people on the benefits and features our applications provide.

Take for example Tripit, illustrated in Figure 2. This application for managing your travel plans by using your travel confirmation emails could easily have asked all new members to sign up through a registration form.

Instead, to the join the service new members simply have to send Tripit a travel confirmation email. From this email, Tripit creates an account and extracts the information it needs to create a rich travel plan for new members. No form required. People sign up for Tripit by using it and learning what the application can do for them.

FIGURE 2. Tripit confirmation email

Gradual engagement is an information architecture challenge because it requires us to think about the core essence of an application and develop a process that introduces it to people while gathering the information the service needs to be useful along the way. This sequencing of information requests as interactions has the potential to create more rewarding and memorable first time experiences for web applications.

Conclusion

By providing a way for designing the questions we are asking, clear paths to completion and gradual engagement processes, information architecture can go a long way toward making web forms more efficient and enjoyable. No wonder there's a lot of IA behind great web forms.

For more on form design, check out Luke's book about Web form usability, visual design and interaction design considerations: Web Form Design: Filling in the Blank *(Rosenfeld Media, 2008).* ■

TOP OF ARTICLE   < PREVIOUS PAGE   NEXT PAGE >