Creating a Design: Interaction

slide 1: Introduction

Hi guys! Welcome to [Fundamentals of User Experience Design ], a Tuts+ premium course. I’m [Sarah Kahn] and today we’re going to talk about designing an Interaction.

slide 2

As you hopefully remember, Interaction Design refers to figuring out ‘what happens next.’ Where do you want users to go, what do you want them to click, what actions actually happen. I’m going to give you some tricks and tips to help you figure all of that out and how to design it.

Here’s what we’re going to cover:

* + Make a map
  + Flow models
  + Whiteboards and brainstorming
  + Assignment

slide 3

[rube golderberg]

For the purposes of this lesson, let’s say you’ve just had a set of requirements handed to you by a client. They need a feature added to the site. Users need to be able to upload a file. Super users need to be notified and be able to approve the file so that it can be posted to a section of the site.

None of the pages currently exist that would allow this to happen. What should you do next?

slide 4

[Make a map]

I’ve generally found it to be the case that interaction design is highly visual. You’re talking about figuring out what happens first, what happens second, except if this exception happens, then it becomes action 2a instead of action number 2. So the best thing to do to get organized is to figure out what actions are involved, and when they happen in relation to each other.

For me, in my lean ux world, I like to do this in a low-tech way on paper or whiteboard. But there are other good tools too.

slide 5

[complicated flow model]

Here’s an example of a large, complex flow model. Someone spent a long time researching user roles, actions, and laying them all out.

Tools that you could use to create something like this would include Microsoft Visio, Omnigraffle, or where I grabbed this from, Cacoo.com. This was one of their example templates. Personally, I would probably not spend time making something this large and complex. But there are a lot of contexts where this would be desirable. If you need to circulate this among a large distributed team or present it to a client, for example.

slide 6

[simple flow model]

now, apologies if you’re having trouble with my chicken scratch handwriting, but this is an example of something that is much more my speed. this is a really quick and dirty flow chart for the requirements we were given earlier. It’s probably not delving in deep enough into the requirements we were given, but it’s a starting place. At this point, i would go back to the stakeholder and say, here’s what I’m thinking. What are we missing? and then you would begin the iterative process of fleshing things out.

slide 7

[how to begin?]

So how do you even being making a flow model like this? Well, you need to look at your requirements, and identify all the actions, and the decision points.

slide 8

[list of requirements]

here’s what we were told.

Regular users need to be able to upload files for approval.

Super users need to be notified that these files are waiting for them, and then they need to approve them or not approve them. Then the files are posted. or not posted.

slide 9

[list of requirements 2]

so reading between the lines, here are some things that we already know here:

before they can do anything, both the user and superuser will have to log in. so we can add that action.

Now, we look for the actions. The phrases I’ve made bold are all actions. This is a good start, but we’re still missing quite a bit here. Here’s where we need to start asking ourselves, What happens next?

slide 10

[holes]

So here are just a few questions we can begin to ask to help us flesh out this process:

* + What does the user see before she has uploaded a file?
  + Can the user see other files she’s uploaded in the past? Can she edit them?
  + How is the superuser notified? Email, RSS feed, in-app messaging?
  + What happens when the superuser rejects a file? Is the user notified?
  + Is the user notified when the file is approved? When it’s posted?

slide 10

[requirements 3]

here I’ve added quite a few different actions to the initial requirements list, after examining the list of questions that we had. This is starting to look more like something that we can get started building. And this still doesn’t even cover all the holes- can a user edit or delete a file before it’s been approved? after it’s been approved? If she can edit before it’s approved but not after, how do we visually indicate that change of state? How do we notify the superuser, if notification is necessary?

And so on. Interaction design is all about examining every possible action, and making sure that everything is covered. You don’t want to forget pieces and leave users without options, or have to cram things in later that might not make as much sense as they would if they had been integrated into the original workflow.

slide 12

[a word about states]

just briefly, i’ve mentioned state a couple of times. i thought we should take a quick detour to define that. when i talk about state, i don’t necessarily mean a separate page, i mean that a given page or piece of a website may be variable.

slide 13

[blank state]

this page is from our example, and it refers to the fact that the page will exist before the user has entered any information by uploading files. it’s better to give some instruction and a call to action, rather than simply displaying an unpopulated list.

slide 14

[a populated state]

here’s that same page, only with a few files listed in it. the same call to action is there, but it’s no longer the only action on the page. we also have some other information, and possibly other actions to take.

slide 15

[app store]

other examples of state include buttons- an enabled button, a disabled button. a really good example of this is in the appstore- so let’s take a quick gander in there and look at these buttons. you can see they say the price of the app, and when you click, they change to an action.

there are lots of instances where the state of an element may change in the backend, and it’s important to maintain consistency and communicate these changes to your users in order to keep things transparent and trustworthy.

slide 16

[whiteboard]

here’s an example of an even rougher, dirtier, but more collaborative approach to the flow model.

this particular picture isn’t of the example we’ve been working with, but it shows the final outcome after a group of 3 or 4 people had been rapidly, verbally, working through the process that we just stepped through. if at all possible, when coming up with the requirements and workflow, it’s great to get a few different heads together. it makes things faster but also you can cover a lot more ground.

slide 17

[polishing things up]

after you’ve got a good idea of what tasks, actions, and states are included, you may want to replace these blocks in your workflow with actual wireframes. this is usually what i end up doing, albeit on paper, so before i send a design to my developers, i have something that looks like this:

slide 14

[assignment]

Alright, time for an assignment!

* + Use the example we’ve been working with. It’s pretty simplistic, but I think the actual functionality would look different on every website.
  + Map out the user flow to add this functionality to \*your\* website.
  + Be sure to identify all states and actions that need to be included.
  + Create a flow model, using paper and pencil or a digital diagraming tool like Cacoo.com

Next time on [Fundamentals of User Experience Design ], will be [Lesson 12: Creating a Design: Wireframing ]. This is [Sarah Kahn], and from all of us here at Tuts+, thanks for listening!