

After defining the project, interviewing stakeholders, and conducting design research, interaction designers next make a series of models, diagrams, and documents. Indeed, this is how designers *design*. Many of these items are paper documents, but they can be physical or digital representations as well. Designers use these models and diagrams to demonstrate their skill, knowledge, and ideas as well as to visualize and analyze what they learned earlier in the design process.

There is a perennial debate about design documentation: how much do you need? Some designers suggest jumping straight into prototyping, saying that far too many documents are created. Others wouldn't dare proceed without most of these documents for fear that something important wasn't written down somewhere.

My answer is that designers need exactly as much documentation as it takes to execute the project well. If the designer's team responds well to use cases, then by all means the designer should produce them. If a client couldn't care less about a task analysis, the designer shouldn't perform one unless the designer personally finds it helpful.

The only reason for interaction designers to make the models and diagrams they do is to communicate their knowledge of and vision for a project. Research models show what was learned from the user research. Personas demonstrate an understanding of the audience. Use cases and task analyses outline what the product or service needs to accomplish. Mood boards, scenarios, storyboards, task flows, sketches, wireframes, and prototypes pull all the pieces together into a vision of what the final product or service will be. Testing makes sure that that vision is shared by the users.

If a document doesn't communicate anything useful, it is worthless—less than worthless, in fact, because it squanders the designer's time. Each model or diagram produced should take the project one step further toward completion. These design documents are at the heart of the craft of interaction design.

The Tools of the Trade

No digital tool has yet to replace the speed and flexibility of a pencil and a piece of paper or a marker and a whiteboard. Those, plus the talent and wisdom of the designer, are the true tools of the trade. Before designers spend

time fiddling with any of the software described here, they should spend some time thinking and planning with pencil and paper or on a whiteboard. Before they use computers to straighten their lines and make everything presentable, designers should rough out what is going to be made. In short, they should design the design.

To make the documents described in this chapter, interaction designers need a working knowledge of several programs in addition to a word processing program:

- ▶ **Diagramming software.** Programs for making charts and diagrams. Microsoft Visio and OmniGraffle are the standards. Adobe InDesign also works well.
- ▶ **Drawing or illustration software.** Programs for making realistic mock-ups of services and products. This software will be used for creating storyboards and prototypes. Adobe Illustrator, Adobe Free-Hand, Adobe Photoshop, and CorelDRAW are all good choices.
- ▶ **Prototyping software.** The software used for prototyping, mainly for digital products. The particular program used will depend on the type of product that the designer is working on. A Web site will require different prototyping software than a physical product probably will. Some typical programs are Adobe Flash, Solidworks, and HTML editors such as BBEdit and Adobe Dreamweaver.
- ▶ **Presentation software.** Programs that designers can use to present their work in front of large groups. Microsoft PowerPoint and Apple Keynote are the standards. Adobe Flash and Acrobat can also be used.
- ▶ **Conversion software.** Programs for creating materials in a format that most people can use. The people a designer deals with may not have the software that the designer is using, so files need to be converted to a readable format. There's really only one option: Adobe Acrobat. Acrobat can convert most files created by the other programs described here to PDF format, which anyone should be able to read.

Research Models

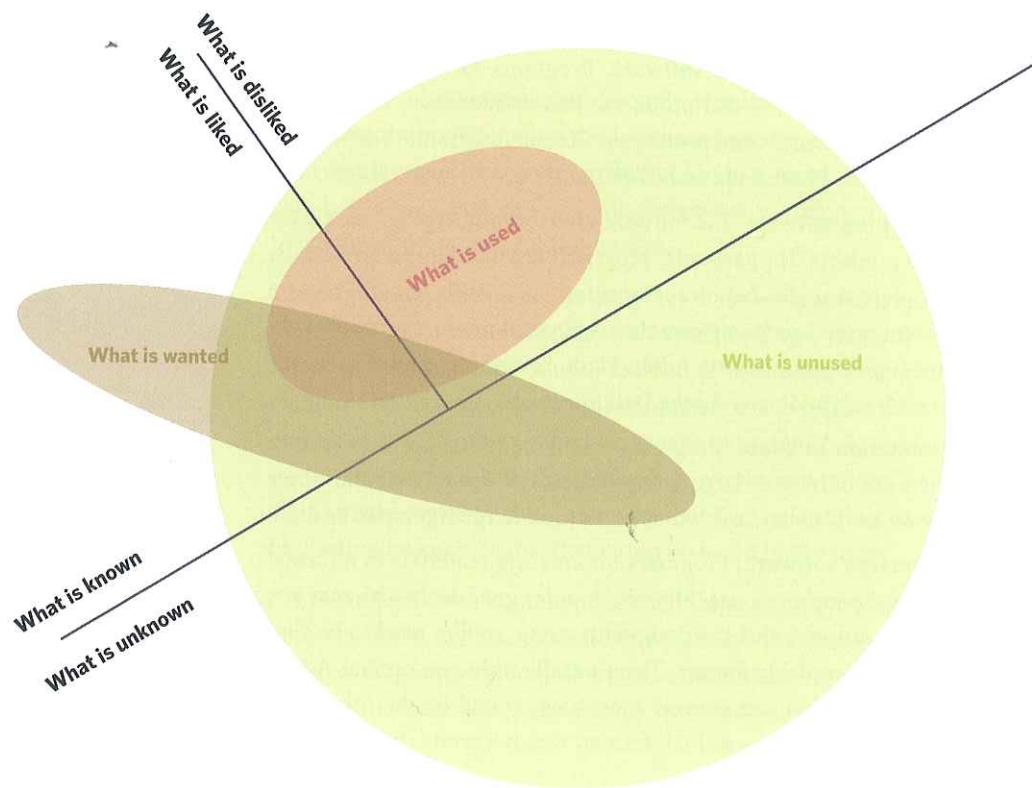
The first models made by designers usually are visualizations of research findings. As discussed in Chapter 4, the designer should have gleaned

Figure 5.1

An example of a model derived from research. Most of the features of the intranet system being researched were unknown and unused, and those that were used were disliked.

design implications from the research data, but some research data—patterns usually—needs to be visualized to be easily (or better) understood.

For example, **Figure 5.1** was created from data collected by talking to users of an intranet. The data revealed that much of the intranet wasn't being used, even features that users said they wanted. Part of the reason these features were unused was because users didn't know they existed—they were buried in the system. The data also showed that most of the features the users did know about, they disliked. Most of what the users wanted in an intranet was unused or unknown or not in the system.



All of this information could, of course, be explained in words, as in the preceding paragraph, or shown in a statistical table. But neither of these has the same impact as the model. Models become design tools, to be referred to repeatedly throughout a project. In this example, the designers could easily see and demonstrate to the client that a key problem is that the users can't find the features they want to use, even when the features already exist in the system.

These are the most common tools for representing research data:

- ▶ **Linear flow.** Shows how a process unfolds over time (**Figure 5.2**). Linear flows are good for showing designers where problems exist in a process.



Figure 5.2
Linear process flow.

- ▶ **Circular flow.** Shows how a process repeats itself in a loop (**Figure 5.3**).

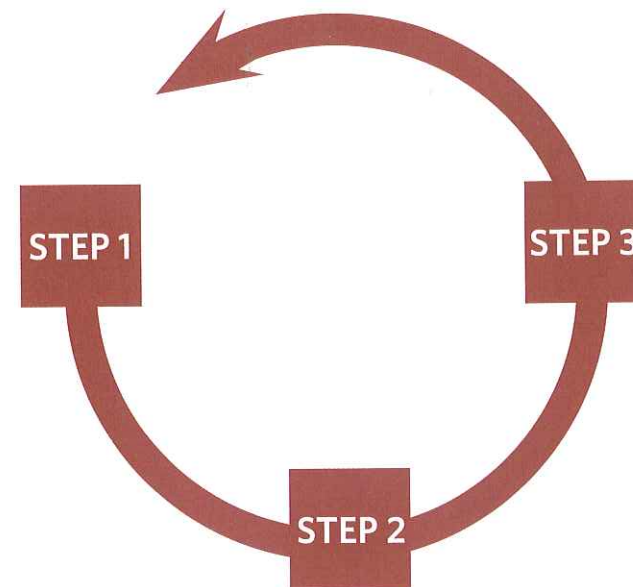
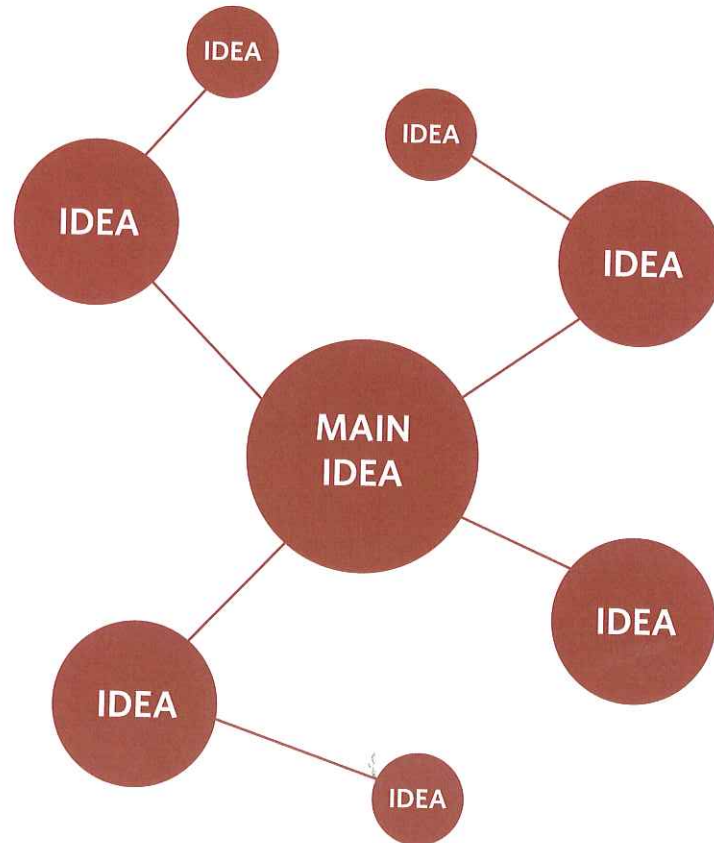


Figure 5.3
Circular process flow.

- ▶ **Spider diagram.** Shows connections between data points. A piece of data is placed in the center of a diagram, and other data radiates out from it (Figure 5.4).

Figure 5.4

Spider diagram.



- ▶ **Venn diagram.** Uses overlapping circles to show relationships and sets (Figure 5.5; Figure 5.1 also is a Venn diagram).

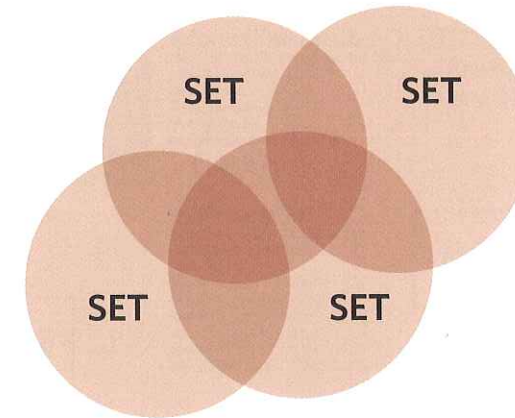


Figure 5.5

Venn diagram.

- ▶ **2x2 matrix.** Shows the relationship between data based on where the data points fall on two axes. These two axes separate data into four quadrants based on two simple variables (Figure 5.6).

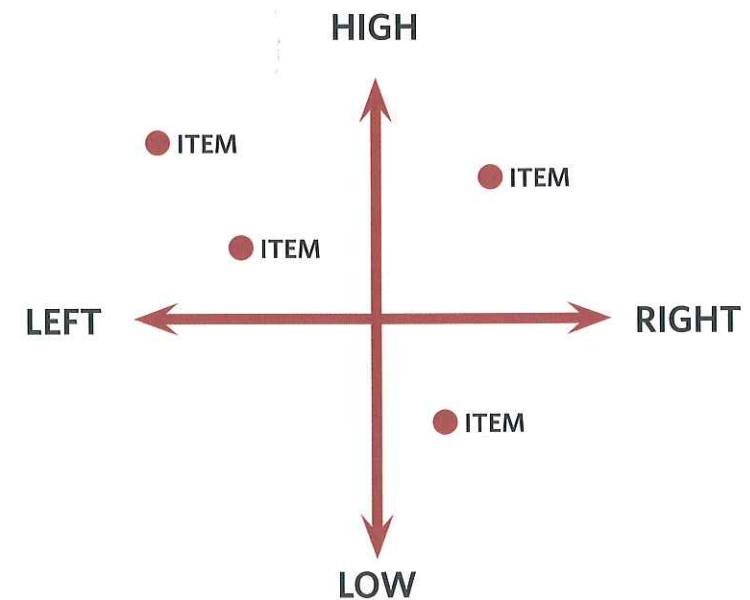


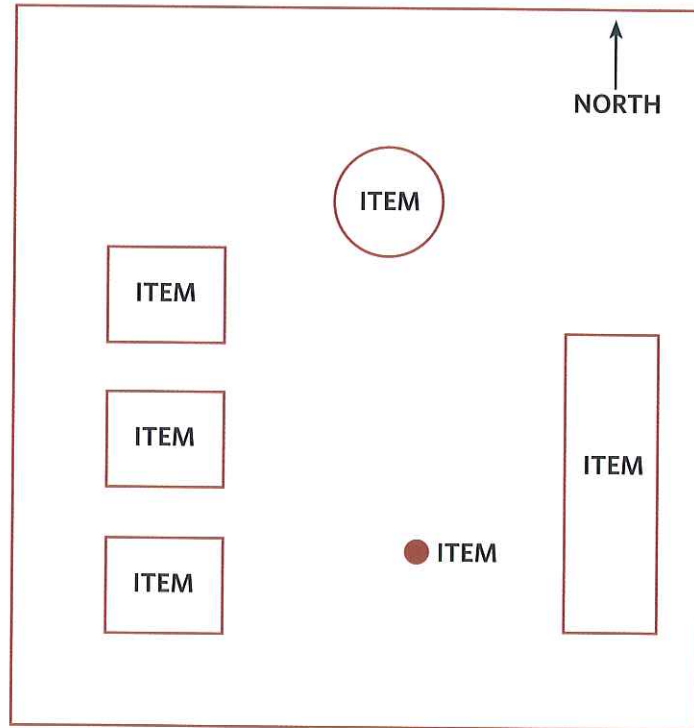
Figure 5.6

2x2 matrix.

Figure 5.7

Map.

- **Map.** Shows spatial relationships (Figure 5.7).



Often, the data itself suggests the correct way to display it. If, for example, the designer observes a step-by-step process that repeats, a circular flow makes sense to use.

Personas

Personas (Figure 5.8) are a documented set of archetypal people who are involved with a product or service. They're supposed to give designers a sense that they are designing for specific people, not just "the users," who, if ill-defined, can be twisted to serve any purpose. ("Of course the users will want to enter a password on every page!")

Figure 5.8

A sample persona. Personas turn "the users" into identifiable human beings.



Dave
the information jockey
primary persona

information usage
info Names, Phone Numbers, Ideas
paper Notebook, Post-Its
info access 3-5x/day
of locations/day 5
% mobile 35
mobile locations Subway, Street

demographics
age 29
occupation Lawyer
location New York City
marital status Single
children None
income \$135,000
education Graduated Law School
hobbies Working Out, Cooking

device usage
computer Sony VIAO Laptop
cell phone Sony Ericsson
pda CLIE
other Network Walkman
primary device Laptop
comfort Comfortable
web 50 hours/week
phone 10 hours/week
programs Email, Word, Excel, IE

"If I'm not connected, I feel like I'm missing something."

Designers should devise personas from observing and talking to users. Personas are typically amalgams of multiple people who share similar goals, motivations, and behaviors. The differences between each persona must be based on these deep characteristics: what people do (actions or projected actions) and why people do them (goals and motivations).

What personas shouldn't be are users who have similar demographics. Focusing on demographics will provide market segments, not personas. The only time demographics really matter for personas is when those demographics directly affect user behavior. A 13-year-old will probably use a product differently than an 83-year-old. A rural peat farmer in Ireland might use a product differently than a Korean financial analyst in Seoul. But maybe not—demographics may not matter at all. In fact, using demographics could limit and hinder the usefulness of the personas. For products with millions of users, for example, a designer could end up with hundreds of personas, and such a large set is essentially useless.

To create a persona, designers find a common set of behaviors or motivations among the people they have researched. This set becomes the basis for the persona, which should be given a name, a picture, and a veneer of demographic data to make the persona seem like a real person.